



ESPE
UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA

**DEPARTAMENTO DE CIENCIAS DE LA ENERGÍA Y
MECÁNICA**

CARRERA DE INGENIERÍA EN MECATRÓNICA

**TRABAJO DE TITULACIÓN, PREVIO A LA OBTENCIÓN DEL
TÍTULO DE INGENIERO EN MECATRÓNICA**

**TEMA: DISEÑO Y CONSTRUCCIÓN DE UN PROTOTIPO
ROBOTIZADO PARA EL DESHIERBE AUTOMÁTICO EN ZONAS
PLANAS DE CULTIVOS DE MAÍZ EN SU ETAPA INICIAL, A
TRAVÉS DE TÉCNICAS DE VISIÓN ARTIFICIAL, PARA LA
ASOCIACIÓN “UNIÓN Y PROGRESO” DEL CANTÓN PÍLLARO,
PROVINCIA DE TUNGURAHUA.**

**AUTORES: MAURICIO DANIEL BARRENO BARRENO
JOSÉ MISAEL JÁCOME BARRIONUEVO**

DIRECTORA: ING. ANDREA CÓRDOVA CRUZATTY. M.Sc.

LATACUNGA

2017



ESPE
UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA

DEPARTAMENTO DE CIENCIAS DE LA ENERGÍA Y MECÁNICA
CARRERA DE INGENIERÍA EN MECATRÓNICA

CERTIFICACIÓN

Certifico que el trabajo de titulación, “**DISEÑO Y CONSTRUCCIÓN DE UN PROTOTIPO ROBOTIZADO PARA EL DESHIERBE AUTOMÁTICO EN ZONAS PLANAS DE CULTIVOS DE MAÍZ EN SU ETAPA INICIAL, A TRAVÉS DE TÉCNICAS DE VISIÓN ARTIFICIAL, PARA LA ASOCIACIÓN “UNIÓN Y PROGRESO” DEL CANTÓN PÍLLARO, PROVINCIA DE TUNGURAHUA**” realizado por los señores **Mauricio Daniel Barreno Barreno** y **José Misael Jácome Barrionuevo**, ha sido revisado en su totalidad y analizado por el software anti-plagio, el mismo que cumple con los requisitos teóricos, científicos, técnicos, metodológicos y legales establecidos por la Universidad de la Fuerzas Armadas ESPE, por lo tanto me permito acreditar y autorizar a los señores **Mauricio Daniel Barreno Barreno** y el señor **José Misael Jácome Barrionuevo** para que lo sustente públicamente.

Latacunga, 15 de diciembre del 2017

A handwritten signature in blue ink that reads 'Andrea Córdova'.

ING. ANDREA CÓRDOVA
DIRECTORA



ESPE

UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA

DEPARTAMENTO DE CIENCIAS DE LA ENERGÍA Y MECÁNICA CARRERA DE INGENIERÍA EN MECATRÓNICA

AUTORÍA DE RESPONSABILIDAD

Nosotros, **Mauricio Daniel Barreno Barreno**, con cédula de ciudadanía N° 180460272-8 y **José Misael Jácome Barrionuevo**, con cédula de ciudadanía N° 180336513-7, declaramos que este trabajo de titulación “**DISEÑO Y CONSTRUCCIÓN DE UN PROTOTIPO ROBOTIZADO PARA EL DESHIERBE AUTOMÁTICO EN ZONAS PLANAS DE CULTIVOS DE MAÍZ EN SU ETAPA INICIAL, A TRAVÉS DE TÉCNICAS DE VISIÓN ARTIFICIAL, PARA LA ASOCIACIÓN “UNIÓN Y PROGRESO” DEL CANTÓN PÍLLARO, PROVINCIA DE TUNGURAHUA**” ha sido desarrollado considerando los métodos de investigación existentes, así como también se ha respetado los derechos intelectuales de terceros considerándose en las citas bibliográficas.

Consecuentemente declaramos que este trabajo es de nuestra autoría, en virtud de ello nos declaramos responsables del contenido, veracidad y alcance de la investigación mencionada.

Latacunga, 15 de diciembre del 2017

A blue ink handwritten signature of Mauricio Daniel Barreno Barreno.

Mauricio Daniel Barreno Barreno
C.C.: 180460272-8

A blue ink handwritten signature of José Misael Jácome Barrionuevo.

José Misael Jácome Barrionuevo
C.C.: 180336513-7



ESPE

UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA

DEPARTAMENTO DE CIENCIAS DE LA ENERGÍA Y MECÁNICA CARRERA DE INGENIERÍA EN MECATRÓNICA

AUTORIZACIÓN

Nosotros, **Mauricio Daniel Barreno Barreno** y **José Misael Jácome Barrionuevo**, autorizamos a la Universidad de las Fuerzas Armadas ESPE publicar en la Biblioteca Virtual de la Institución el presente trabajo de titulación **“DISEÑO Y CONSTRUCCIÓN DE UN PROTOTIPO ROBOTIZADO PARA EL DESHIERBE AUTOMÁTICO EN ZONAS PLANAS DE CULTIVOS DE MAÍZ EN SU ETAPA INICIAL, A TRAVÉS DE TÉCNICAS DE VISIÓN ARTIFICIAL, PARA LA ASOCIACIÓN “UNIÓN Y PROGRESO” DEL CANTÓN PÍLLARO, PROVINCIA DE TUNGURAHUA”** cuyo contenido, ideas y criterios son de nuestra autoría y responsabilidad.

Latacunga, 15 de diciembre del 2017

Una firma manuscrita en tinta azul que parece decir 'Mauricio Daniel Barreno Barreno'.

Mauricio Daniel Barreno Barreno
C.C.: 180460272-8

Una firma manuscrita en tinta azul que parece decir 'José Misael Jácome Barrionuevo'.

José Misael Jácome Barrionuevo
C.C.: 180336513-7

DEDICATORIA

Mauricio Daniel Barreno Barreno

A mis padres Levi Gabriel Barreno y Teresa Barreno, quienes a pesar de todo me han apoyado incondicionalmente para que me pueda defender en la vida.

A mis dos hermosas hermanas Paulina y Dayanna, a quienes quiero muchísimo.

A todas las personas que me han ayudado y motivado para que pudiera formarme profesionalmente.

José Misael Jácome Barrionuevo

A mis padres Misael y Victoria por hacer todo lo que soy, mi hermano Andrés y familiares que han podido brindarme su apoyo y preocupación a su medida durante todo el trayecto de la carrera.

A mis amigos y compañeros quienes ofrecieron su ayuda y aliento para conseguir con este importante objetivo.

Al Todopoderoso por permitirme vivir este momento.

AGRADECIMIENTO

A Dios por darnos la capacidad y oportunidad de estudiar, la paciencia y la motivación para nunca rendirnos ante las adversidades que se presentaron a lo largo de este trayecto.

A nuestras familias, por ese apoyo incondicional para en cada paso que dimos en la Universidad de las Fuerzas Armadas, por sus palabras de aliento ante los problemas que nunca faltaban cada semestre, y por su paciencia que nos ha permitido culminar exitosamente nuestra etapa de formación profesional.

Nuestros agradecimientos de manera especial a la ingeniera Andrea Córdova, por su guía, su ayuda, su motivación y por sus palabras de aliento que nunca faltaron por su parte.

A la Asociación de Agricultores “Unión y Progreso” perteneciente al cantón Píllaro por brindarnos su apoyo total en la elaboración de este proyecto

Por ultimo pero no menos importante, a todas y cada una de las personas que contribuyeron de una u otra manera a la realización de este proyecto.

ÍNDICE DE CONTENIDOS

CARÁTULA	i
CERTIFICACIÓN	ii
AUTORÍA DE RESPONSABILIDAD	iii
AUTORIZACIÓN	iv
DEDICATORIA	v
AGRADECIMIENTO	vi
ÍNDICE DE CONTENIDOS	vii
ÍNDICE DE FIGURAS	xii
ÍNDICE DE TABLAS	xviii
RESUMEN	xx
ABSTRACT	xxi
CAPÍTULO I	1
ASPECTOS GENERALES	
1.1 Planteamiento del problema	1
1.2 Antecedentes.....	3
1.3 Justificación e Importancia	5
1.4 Hipótesis.....	6
1.5 Objetivos	6
1.5.1 Objetivo General.....	6
1.5.2 Objetivos Específicos	7
1.6 Alcance.....	7
CAPÍTULO II	8
MARCO TEÓRICO	
2.1 Cultivo de Maíz.....	8
2.1.1 Maíz.....	8
2.1.2 Desarrollo vegetativo del Maíz	8
2.1.3 Siembra	9

2.1.4	Deshierba del maíz.....	9
2.1.5	Tipos de Deshierba	10
2.1.6	Tipos de herramientas mecánicas de deshierba	11
2.1.7	Tecnología automática en la eliminación de malas hierbas	15
2.2	Componentes de Robots.....	15
2.2.1	Actuadores eléctricos	15
2.2.2	Ruedas	18
2.2.3	Sensores	19
2.2.4	Sistemas de alimentación.....	21
2.2.5	Microcontroladores	22
2.2.6	Sistemas de suspensión.....	24
2.2.7	Unidad de Control.....	25
2.3	Robótica Móvil.....	26
2.3.1	Vehículos con ruedas	26
2.3.2	Navegación de Robots Móviles	28
2.3.3	Localización de Robots Móviles	29
2.3.4	Dead Reckoning	29
2.4	Robots Paralelos	30
2.4.1	Definición.....	30
2.4.2	Robot tipo Delta.....	31
2.4.3	Configuraciones de los Robots tipo Delta.....	31
2.4.4	Cinemática del robot tipo Delta.....	32
2.5	Visión Artificial	33
2.5.1	Imagen Digital.....	33
2.5.2	Etapas de un sistema de Visión Artificial.....	34
2.6	Aprendizaje Profundo (Deep Learning)	37
2.6.1	Neurona Artificial	37
2.6.2	Redes Neuronales Artificiales.....	38
2.6.3	Aprendizaje de las Redes Neuronales	38
2.6.4	Redes Neuronales Convolucionales.....	41

CAPÍTULO III 44

DISEÑO MECÁNICO DEL PROTOTIPO

3.1	Diseño del Robot Delta.....	44
3.1.1	Selección de los componentes del robot Delta.....	47
3.1.2	Eector final.....	54
3.1.3	Selección de los servomotores del Robot Delta	62
3.2	Diseño de la estructura de la plataforma	64
3.2.1	Selección del tipo de configuración del robot móvil	65
3.2.2	Estructura superior de la plataforma móvil	66
3.2.3	Ruedas	69
3.2.4	Rueda Loca	71
3.2.5	Sistema de suspensión	71
3.2.6	Selección de los motores	77
3.3	Ensamblaje del Prototipo.....	81
3.3.1	Construcción de las ruedas	82
3.3.2	Construcción del sistema de suspensión y acople a plataforma	84
3.3.3	Construcción de la plataforma móvil.....	86
3.3.4	Construcción del robot Delta con el eector final	88
3.3.5	Ensamblaje total	91

CAPÍTULO IV 94

DESARROLLO DEL CONTROL DEL PROTOTIPO

4.1	Selección de componentes para el control.....	94
4.1.1	Selección del microcomputador para visión	95
4.1.2	Selección de la cámara	97
4.1.3	Selección del microcontrolador.....	101
4.1.4	Selección del controlador de los motores principales.....	103
4.1.5	Selección del encoder para los motores principales.....	104
4.1.6	Selección de las baterías	105
4.1.7	Selección del sistema de comunicación	108

4.1.8 Selección del software.....	109
4.1.9 Circuitos impresos adicionales y otros componentes.....	110
4.2 Diseño del controlador para la detección de mala hierba.....	116
4.2.1 Selección del modelo de machine learning	116
4.2.2 Selección del framework	118
4.2.3 Preparación de la red	119
4.2.4 Selección y entrenamiento de la arquitectura de red.....	123
4.2.5 Clasificación	125
4.3 Diseño del controlador para establecimiento de la trayectoria	132
4.3.1 Generación de hileras	133
4.3.2 Transmisión de los datos.....	137
4.3.3 Control de localización.	139
4.4 Diseño del controlador para la eliminación de mala hierba	145
4.4.1 Selección de datos para el robot Delta	145
4.4.2 Transmisión de datos	149
4.4.3 Diseño del Control del Robot Delta	149

CAPÍTULO V **157**

PRUEBAS Y RESULTADOS

5.1 Descripción del campo de prueba	157
5.2 Pruebas del funcionamiento prototipo	159
5.2.1 Detección y clasificación	159
5.2.2 Navegación.....	163
5.2.3 Robot Delta.....	165
5.2.4 Deshierba	171
5.3 Validación de la Hipótesis	178
5.3.1 Hipótesis nula y alternativa.....	178
5.3.2 Observaciones.....	178
5.3.3 Prueba de bondad de ajuste	179
5.3.4 Grados de libertad	180
5.4 Limitaciones.....	181

5.5 Costos	182
CAPÍTULO VI	184
CONCLUSIONES Y RECOMENDACIONES	
6.1 Conclusiones.....	184
6.2 Recomendaciones.....	186
REFERENCIAS BIBLIOGRÁFICAS.....	189
ANEXOS	198
CERTIFICACIÓN	199

ÍNDICE DE FIGURAS

Figura 1 Persona quitando la mala hierba en un cultivo de maíz.....	2
Figura 2 Robots autónomos de deshierba actuales	5
Figura 3 Etapas del desarrollo del cultivo de maíz	8
Figura 4 Distribución del maíz en la Siembra	9
Figura 5 Cultivo de una leguminosa con alta presencia de maleza.....	10
Figura 6 Deshierbe tipo manual de maíz en Píllaro.....	10
Figura 7 Deshierbe por herbicidas	11
Figura 8 Herramienta de Deshierbe tipo Canasta	12
Figura 9 Aplicación del Deshierbe de dedos	12
Figura 10 Desherbado con la Herramienta de Torsión.....	13
Figura 11 Herramienta rotatoria de desherbado por cepillos	14
Figura 12 Eco-weeder con Manipulación Humana y Accionamiento Manual.....	14
Figura 13 Partes de un motor DC convencional.....	16
Figura 14 Rotor y Estator en un motor DC	17
Figura 15 Vista frontal y lateral interna de un servomotor	17
Figura 16 Tren de pulsos para el control del ángulo de un servomotor.....	18
Figura 17 Rueda de Carretilla	18
Figura 18 Rueda loca	19
Figura 19 Diagrama típico de un encoder	20
Figura 20 Sensores externos	21
Figura 21 Batería de Plomo y Ácido a 12V y 20Ah	22
Figura 22 Microcontrolador PIC16F628A	23
Figura 23 Partes de un sistema de suspensión.....	24
Figura 24 Sistema Ackerman	26
Figura 25 Triciclo Clásico	27
Figura 26 Sistema de direccionamiento diferencial	27
Figura 27 Otras configuraciones	28
Figura 28 Trayecto establecido por dead reckoning.....	29
Figura 29 Odometría de un robot móvil.....	30

Figura 30 Robot Delta clásico	31
Figura 31 Configuraciones del robot Delta	32
Figura 32 Pixeles en una imagen RGB y en escala de grises.....	34
Figura 33 Proceso de captura de una Imagen Digital	35
Figura 34 Resolución en la representación de una imagen	35
Figura 35 Ejemplo de la utilización de un filtro	36
Figura 36 Imagen de una planta, con segmentación canal verde	36
Figura 37 Representación simplificada de una neurona artificial	37
Figura 38 Red neuronal común de cuatro capas y sus partes constitutivas.....	38
Figura 39 Descenso del gradiente.....	39
Figura 40 Diagrama típico de una Red Neuronal Convolutiva.....	41
Figura 41 Operación de Convolución sobre una matriz	42
Figura 42 Función de Pooling o Subsampling, con valores al azar	42
Figura 43 Arquitectura de Red Neuronal Convolutiva <i>cNET</i>	43
Figura 44 Algoritmo del cálculo de dimensiones óptimas del robot Delta	45
Figura 45 Volumen de trabajo del robot Delta	46
Figura 46 Vista del plano XZ y XY del volumen de trabajo del robot Delta	47
Figura 47 Diseño CAD del eslabón corto.	47
Figura 48 Análisis de Von Mises del eslabón corto del robot Delta.....	49
Figura 49 Factor de Seguridad del eslabón pequeño del robot Delta	49
Figura 50 Eslabón Largo (Varilla roscada m4) y Tubo de Aluminio de 5/16.....	50
Figura 51 Diseño CAD de la base superior.	51
Figura 52 Análisis de Von Mises de la base fija del robot Delta	51
Figura 53 Factor de seguridad de la base fija del robot Delta	52
Figura 54 Diseño CAD de la base móvil.....	52
Figura 55 Análisis estático de la base fija del robot Delta	53
Figura 56 Junta esférica m4 Traxxas 5347	54
Figura 57 Modelo de la herramienta de corte.....	56
Figura 58 Detalle del Conjunto Motor - Reducción Motores Syma X8C.....	60
Figura 59 Análisis estático la herramienta del robot Delta	61
Figura 60 Torques en eslabón 1 y el Servomotor.....	62

Figura 61 Servomotor Tower Pro MG996R.....	64
Figura 62 Ensamble del robot tipo Delta	64
Figura 63 Diseño de la estructura superior del prototipo	66
Figura 64 Análisis estático de la estructura superior de la plataforma.	67
Figura 65 Detalle del soporte en la estructura del prototipo	68
Figura 66 Muestra de la carcasa exterior del prototipo	69
Figura 67 Detalle de la rueda montada con la estructura de suspensión	69
Figura 68 Base para sensor óptico con sujeciones	70
Figura 69 Detalle del disco ranurado.....	70
Figura 70 Rueda Loca giratoria y su acople para la estructura superior	71
Figura 71 Masa suspendida y masa no suspendida en un sistema móvil.	72
Figura 72 Medidas utilizadas para calcular el índice de movimiento	73
Figura 73 Suspensión Ke Zhen 750	74
Figura 74 Sistema de suspensión delantera del prototipo.....	75
Figura 75 Análisis de Von Mises del Sistema de Suspensión	76
Figura 76 Factor de Seguridad del Sistema de Suspensión.	76
Figura 77 Diagrama de cuerpo libre de la llanta.....	78
Figura 78 Motor Banebots RS775 18V y su caja reductora	81
Figura 79 Ensamblaje del conjunto motor-caja reductora	82
Figura 80 Construcción del bocín.....	82
Figura 81 Proceso de corte láser en baquelita	83
Figura 82 Disposición final de los componentes de la rueda	84
Figura 83 Construcción del sistema de suspensión	85
Figura 84 Detalle de la rueda loca.....	85
Figura 85 Construcción de la Estructura de la plataforma móvil.	86
Figura 86 Detalle del soporte de la cámara.....	87
Figura 87 Construcción del Recubrimiento.....	87
Figura 88 Proceso de corte por plasma sobre aluminio	88
Figura 89 Disposición del acople de los servomotores	88
Figura 90 Acople eslabón corto con el servomotor	89
Figura 91 Detalle ensamblaje eslabón corto y largo.....	89

Figura 92 Detalle subensamblaje Eslabón Largo - Base inferior.....	90
Figura 93 Detalle de la disposición de la herramienta.....	90
Figura 94 Detalle ensamblaje con la plataforma móvil.....	91
Figura 95 Robot Delta ensamblado en la plataforma.....	91
Figura 96 Soporte ensamblado en la estructura.....	92
Figura 97 Ensamble final del prototipo.....	93
Figura 98 Microcomputador Raspberry Pi 3.....	97
Figura 99 Detalle del Módulo de Cámara V2 para Raspberry Pi.....	97
Figura 100 Campo de visión estimado para la cámara.....	99
Figura 101 Relación de la longitud focal con la altura de la cámara.....	100
Figura 102 Medidas de ubicación de la cámara en el soporte cámara.....	101
Figura 103 Campo de visión final de la cámara.....	101
Figura 104 Batería sbb AGM 20Ah / 12VDC.....	107
Figura 105 Batería Externa ARGOMTech 10000mAh.....	108
Figura 106 Librerías usadas para cada dispositivo.....	110
Figura 107 Fuente de 6V 3A.....	111
Figura 108 Circuito de control para la herramienta.....	112
Figura 109 Circuito para elementos de control.....	113
Figura 110 Circuito de Shield para Arduino.....	114
Figura 111 Esquemático del circuito final.....	115
Figura 112 Disposición final de los componentes.....	116
Figura 113 Proceso de Segmentación de las Muestras.....	120
Figura 114 Proceso de entrenamiento en Linux.....	122
Figura 115 Detalle de la red cNet.....	124
Figura 116 Proceso de entrenamiento de la red usando el Solver Adam.....	125
Figura 117 Diagrama de Bloques del Proceso de Adquisición de imágenes.....	126
Figura 118 Diagrama de bloques del proceso de Pre-Procesamiento.....	127
Figura 119 Error de binarización.....	128
Figura 120 Diagrama de flujo la reconstrucción de contornos.....	129
Figura 121 Contornos reconstruidos para los casos anteriores.....	130
Figura 122 Diagrama de Bloques de la Etapa Segmentación.....	131

Figura 123 Diagrama de procesos de clasificación con una planta de maíz.....	131
Figura 124 Estructura y vectores para almacenar los datos de clasificación	132
Figura 125 Diagrama de flujo de búsqueda de extremos.....	134
Figura 126 Ejemplo de la implementación del algoritmo anterior	135
Figura 127 Detalle del algoritmo cuando sólo hay una hilera visible	136
Figura 128 Cálculo del centro en dos hileras (Círculo negro)	136
Figura 129 Centro para una única hilera (Círculo negro)	137
Figura 130 Diagrama de transmisión del centro de la hilera	139
Figura 131 Parámetros de odometría de un robot diferencial	140
Figura 132 Algoritmo para cálculo de Odometría	141
Figura 133 Visualización de los errores en X y en Y	142
Figura 134 Visualización del error en la dirección del prototipo	143
Figura 135 Diagrama de bloques de un control proporcional.....	143
Figura 136 Diagrama de flujo del control proporcional para la navegación.....	144
Figura 137 Gráficas del control proporcional con distintas constantes Kp	145
Figura 138 Área trabajo del robot Delta	146
Figura 139 Ciclo de procesamiento para la actuación del robot Delta	147
Figura 140 Arriba: Datos totales Abajo: Datos necesarios	148
Figura 141 Diagrama de bloques de la comunicación.....	149
Figura 142 Modelo de la geometría de un Robot Delta.....	150
Figura 143 Ángulos de una cadena cinemática del robot Delta	151
Figura 144 Representación vectorial de la primera cadena cinemática	153
Figura 145 Diagrama de flujo para el control del robot Delta	155
Figura 146 Diagrama de control total del prototipo.....	156
Figura 147 Ubicación del terreno de maíz.....	157
Figura 148 Izquierda: Maíz V7, Derecha: Maíz V10.....	158
Figura 149 Plantas del terreno de cultivo	158
Figura 150 Distribución de los tiempos de procesamiento	160
Figura 151 Proceso de reducción de contornos	161
Figura 152 Porcentaje de reducción de contornos	163
Figura 153 Pruebas de navegación en el terreno de cultivo	164

Figura 154 Robot Delta recorriendo el espacio de trabajo	169
Figura 155 Comparación de la clasificación maíz V7	173
Figura 156 Frecuencia de deshierba en maíz V7	175
Figura 157 Comparación de la clasificación maíz V10	176
Figura 158 Frecuencia de deshierba en maíz V10	177
Figura 159 Distribución de Chi Cuadrado con los valores obtenidos	181

ÍNDICE DE TABLAS

Tabla 1 Estudio del Rendimiento en el Deshierbe a diferentes métodos	2
Tabla 2 Funciones de activación para redes neuronales	40
Tabla 3 Dimensiones de las partes del robot Delta	46
Tabla 4 F.S. para diseño de flechas según la norma ANSI-ASME.....	48
Tabla 5 Distribución de las herramientas comerciales de deshierba.....	55
Tabla 6 Valores y direcciones de identificación de S_c para suelo arcilloso	57
Tabla 7 Descripción de las características técnicas del Motor Syma X8C	59
Tabla 8 Valores promedios obtenidos de parámetros eléctricos del motor	60
Tabla 9 Matriz de Selección de los Servomotores para el Robot Delta.....	63
Tabla 10 Matriz de selección de la disposición de la plataforma móvil	65
Tabla 11 Matriz para Selección del tipo Motores del Prototipo	77
Tabla 12 Selección de los Motores a la potencia calculada	80
Tabla 13 Elementos para realizar el control del prototipo	94
Tabla 14 Características de tarjetas Controladoras	95
Tabla 15 Características Técnicas del Módulo de Cámara V2.....	98
Tabla 16 Características de microcontroladores populares	102
Tabla 17 Características de tarjetas controladoras de motores	103
Tabla 18 Características de sensores infrarrojos	104
Tabla 19 Consumo energético del prototipo.....	106
Tabla 20 Características de la Batería sbb AGM 20Ah / 12VDC.....	107
Tabla 21 Protocolos de comunicación para los elementos de control.....	109
Tabla 22 Comparativa entre distintos modelos de Machine Learning	117
Tabla 23 Comparativa de Frameworks de Software Libre	118
Tabla 24 Comparación técnica de las arquitecturas seleccionadas	123
Tabla 25 Comparación del entrenamiento con distintos tipos de solvers.....	124
Tabla 26 Tiempo de procesamiento por etapa	159
Tabla 27 Reducción de contornos por cada etapa	162
Tabla 28 Pruebas de navegación en superficie lisa	163
Tabla 29 Pruebas de navegación en el terreno de cultivo (en mm.....	165

Tabla 30 Pruebas de posicionamiento del robot Delta	166
Tabla 31 Diferencias entre los ángulos de los servomotores	167
Tabla 32 Puntos extremos y ángulos de los servos del robot Delta	168
Tabla 33 Valores del posicionamiento del robot Delta	170
Tabla 34 Pruebas de clasificación para Maíz V7.....	173
Tabla 35 Pruebas de deshierba sobre áreas de trabajo de maíz V7.....	174
Tabla 36 Pruebas de clasificación para Maíz V10.....	176
Tabla 37 Pruebas de deshierba sobre áreas de trabajo de maíz V10	177
Tabla 38 Tabla de valores observados	179
Tabla 39 Distribución de valores esperados	180
Tabla 40 Descripción de costos del desarrollo del prototipo	182

RESUMEN

La agricultura en el Ecuador representa una de las actividades económicas más importantes, siendo el maíz uno de los cultivos ancestrales y de mayor consumo en el país. La deshierba del maíz es una acción necesaria para su correcto desarrollo en su etapa inicial, esta actividad se ha venido dando mayormente de manera manual pero en la actualidad es tecnológicamente posible desarrollar un robot que cumpla esta función. Para lo cual se propone el diseño e implementación de un prototipo robotizado de deshierba para campos de maíz en su etapa inicial. El proyecto se compone de 4 subsistemas: un brazo robótico paralelo de tipo Delta, el cual tiene por función el posicionamiento de la herramienta de eliminación sobre las malas hierbas; una plataforma móvil de tipo diferencial dotada de un sistema de suspensión que posee un sistema de control de la navegación realizado en base de técnicas de odometría; un sistema de visión artificial que conjuntamente con una Red Neuronal Convolutiva se encarga de adquirir las imágenes del cultivo, ubicar y clasificar las plantas en maíz y mala hierba , así como generar la trayectoria para la navegación y el área de trabajo, finalmente un panel de control externo para manipular el prototipo. Todos estos sistemas se comunican a través de un protocolo de comunicación tipo serial RS232. La implementación de este proyecto representa un apoyo para los agricultores de la Asociación “Unión y Progreso” del cantón Píllaro que es la institución beneficiaria del prototipo.

PALABRAS CLAVE:

- **AGRICULTURA DE PRECISIÓN**
- **MAÍZ - CULTIVO**
- **ROBOT DELTA**
- **VISIÓN ARTIFICIAL**
- **REDES NEURONALES CONVOLUCIONALES**

ABSTRACT

Agriculture in Ecuador represents one of the most important economic activities, with corn being one of the ancestral and most consumed crops in the country. Weeding maize is a necessary action to allow the correct development of the plant in its initial stage, that activity has been done mostly manually but nowadays its technological possible to develop a robot able to achieve this function. Therefore, in this document it has purposed a design and implementation of a prototype of a weeding robot for maize fields in its initial stages. The project is composed by four subsystems: a Delta parallel robot, which has the function of positioning the elimination tool on the weed plants; a mobile platform of differential drive equipped with a suspension system that has a control system of navigation based on odometry techniques; an artificial vision system that works together with a Convolutional Neural network to get images of the crop, locating and classify the plants of maize and weed, as well as generating the trajectory to the mobile platform and the work area, finally a external control panel to manipulate the prototype. A RS232 Serial Communication Protocol communicates all these systems. The implementation of this project represents a support for the farmers of the Association “Union y Progreso” of Píllaro Town.

KEYWORDS:

- **PRECISION AGRICULTURE**
- **MAIZE CROP**
- **DELTA ROBOT**
- **ARTIFICIAL VISION**
- **CONVOLUTIONAL NEURAL NETWORKS**

CAPÍTULO I

ASPECTOS GENERALES

1.1 Planteamiento del problema

Una de las principales preocupaciones en la actualidad para Ecuador y el mundo es la sanidad de los alimentos provenientes del campo, el abuso de sustancias químicas en los cultivos ha hecho que sea dudosa la salubridad de los mismos, además que pueden contaminar fuentes de agua, generando daños en algunas especies de plantas nativas e incluso en la salud de las personas que cultivan o consumen dichos alimentos.

En vista de eso, muchos agricultores están adoptando técnicas para cultivos orgánicos, los cuales están teniendo más demanda en mercados nacionales e internacionales, además el Estado Ecuatoriano, analiza y aprueba leyes que promuevan acciones para que la población y en especial los niños, cuenten con alimentos sanos para que de esta forma se pueda alcanzar una adecuada nutrición.

Por lo tanto, el objetivo de la Asociación “Unión y Progreso” es lograr que sus miembros vayan adoptando los postulados de la soberanía alimentaria y aumenten sus ingresos al aplicar nuevas y mejores técnicas en el campo de la agricultura. Debido a que se busca promover una producción más limpia, actualmente existen mayores controles de calidad respecto a los químicos en los alimentos, de donde surge la necesidad de usar técnicas que permitan alcanzar los nuevos estándares de calidad en los productos agrícolas.

Una de las principales sustancias químicas utilizadas en la agricultura para la eliminación de malas hierbas o malezas son los herbicidas, los cuales se pueden reducir o eliminar al usar nuevas tecnologías como la robótica.

La acción denominada como *deshierba* o *escarda* (que será capaz de realizar el prototipo) trae también sus beneficios en el rendimiento de la

producción como muestra la siguiente tabla sobre el rendimiento del maíz forrajero.

Tabla 1
Estudio del rendimiento en el deshierbe a diferentes métodos

Tratamiento	Rendimiento(kg/ha ms)
Testigo	11 782
Térmico	11 690
Químico	13 648
Mecánico	13 759
Manual	16 217

Fuente: (Suárez, Piñeiro, & Valladares, 2005)

Como se muestra en la anterior tabla, el deshierbe manual es un proceso muy difícil y costoso al tener que contratar varias personas. Por lo cual al implementar un sistema mecánico económico y acompañado por un buen sistema de detección de malas hierbas puede proveer una mejor eliminación de la maleza propia del maíz y ser más accesible al bolsillo del agricultor.

También se puede aportar con la mejora de la calidad de vida para las personas dedicadas a la agricultura, debido a que en la época de deshierba, estas se exponen a la radiación solar y al polvo, además de sufrir de dolores en su parte lumbar por la mala posición que mantienen por el uso prolongado de herramientas como el azadón, dicha acción se puede apreciar en la siguiente figura.



Figura 1 Persona quitando la mala hierba en un cultivo de maíz.

Fuente: (Budabar, 2015)

1.2 Antecedentes

La agricultura en la historia de la humanidad, siempre ha sido el pilar fundamental para el desarrollo de nuestra civilización. Desde los primeros asentamientos humanos y hasta nuestros días, el manejo de la agricultura se ha conservado como una labor mayormente manual.

Se han tenido avances graduales para lograr una mejor producción y en los últimos años se ha buscado una obtención de alimentos más ecológicos para asegurar la sostenibilidad alimentaria del futuro.

El aumento de la demanda de alimentos por el crecimiento poblacional ha necesitado de una mayor producción de cultivos, sin embargo, un problema para la sostenibilidad alimentaria es la poca mano de obra para esta labor y su progresiva disminución. En América Latina para el año 2015 se estimaba que la población urbana fue del 80% (UNFPA Ecuador, 2014), tal desbalance entre la población consumidora y productora hace que sea necesario proponer alternativas ecológicas y eficientes que permitan asegurar el abastecimiento de productos agrícolas.

El maíz, sentli, sara, jank'a o avati es uno de los cultivos más importantes en América Latina desde el punto de vista cultural, social, económico y alimenticio (Rallt, 2017). Desde su aparición en América hasta la actualidad, ha ganado una gran importancia en la población mundial, pues este cereal es base de dietas para una multitud de pueblos a lo largo del mundo y sus derivados son muy apreciados en el sector industrial.

En el cultivo del maíz se debe realizar varios controles de malas hierbas desde la siembra hasta antes de terminar la etapa de desarrollo vegetativo (previo a la floración), que se da aproximadamente a los 55 días a partir de la siembra, puesto que después de esta etapa se realiza una técnica de desplazamiento de tierra hacia la planta, denominada aporque. El (SINAGAP, 2015), recomienda

hacer 5 deshierbas antes de terminar el desarrollo vegetativo, para una mejor producción del cultivo, el deshierbe se le puede hacer de manera manual o química al utilizar sustancias como herbicidas.

Desde la invención del primer herbicida, si bien ha habido una mayor producción agrícola, los impactos ambientales son notorios desde el progresivo daño al suelo hasta la afectación a la salud humana. Esto ha hecho que los agricultores busquen soluciones amigables para el ambiente. Los métodos mecánicos son una alternativa ecológica al deshierbe de parcelas pero su precisión y su costo al requerir maquinaria pesada no lo hace accesible a todos los agricultores.

A inicios del Siglo XXI múltiples instituciones han ideado y creado diferentes soluciones robóticas para la agricultura. Una de las más viables es el deshierbado robotizado. El avance en el deshierbado mecatrónico en Europa y en Asia ha sido notorio asociado principalmente a la Agricultura Orgánica y de Precisión, principalmente enfocado a sus cultivos de mayor frecuencia como de arroz, maíz y legumbres.

Uno de los primeros avances en este campo fue el proyecto denominado “Segmentation of plants and weeds for a precision crop protection robot using infrared images” (Brivot & Marchant, 1996) quienes se encargaron de iniciar el proceso a través de luz infrarroja para la segmentación de plantas en cultivos distribuidos por filas para diferenciarlas de la mala hierba.

A su vez uno de los primeros proyectos robotizados para la eliminación de malas hierbas en cultivos fue el denominado “Mobile Robot for Weeding” (Ertbølle & Lavdal, 2001) destinado a la eficiencia en la agricultura bajo el lema de usar menos herbicidas con el desarrollo de un prototipo autónomo que controla la formación de mala hierba para una mejor producción en los cultivos.

En los últimos años se han desarrollado proyectos muy exitosos como los casos del Robot Ecorobotix (Ecorobotix, 2017) que fue iniciado en Suiza para

realizar el deshierbe autónomo en cultivos a través de una estructura en base de paneles solares que ofrecen una autonomía muy alta para grandes cultivos. Otra historia de éxito desarrollada en Francia es el Robot Autónomo OZ (Naïo Technologies, 2013) el mismo que con su sistema autónomo de manejo se puede acoplar a cualquier tipo de cultivo con un tiempo bajo de entrenamiento para que el robot funcione.



(a) Naïo Oz Robot¹ (b) Ecorobotix²

Figura 2 Robots autónomos de deshierba actuales

Fuente: ¹ (Naïo Technologies, 2013), ² (Ecorobotix, 2017)

La Asociación de Agricultores “Unión y Progreso” del Cantón Píllaro es una organización fundada en los años 90, su principal actividad es el desarrollo y cultivo de productos agrícolas, entre los más importantes se destacan el maíz, la papa y frutas, la asociación se creó para permitir a sus miembros la entrega de sus productos directamente a las empresas procesadoras de alimentos, entre ellas Planhofa de la Ciudad de Ambato.

1.3 Justificación e Importancia

En Ecuador poco a poco la robótica está teniendo acogida en la agricultura, muestra de ello son los robots aéreos no tripulados (drones) que ya están siendo utilizados en algunas provincias de la Costa ecuatoriana para el monitoreo y

fumigaciones de cultivos como banana y caña de azúcar, con el fin de mejorar las condiciones de los cultivos (Peñañiel, 2015).

El prototipo robotizado de desherbado o eliminador de malas hierbas en el cultivo de maíz a implementar, posee una gran ventaja ya que no utilizará herbicidas, como lo hacen muchos de los robots comerciales de este tipo. Evitando así la aparición de malas hierbas resistentes por el abuso en la utilización de herbicidas como el glifosato y paraquak, estos son los matamalezas más utilizados en Ecuador, son altamente tóxicos y su acción contaminante afecta no solamente al suelo sino también la salud del ser humano. (Nivia, 2003)

Finalmente, otro aspecto que se debe tratar es la mejora en la calidad de vida de los agricultores, con este tipo de prototipo se busca que estas personas requieran de menos esfuerzo físico para la realización de la deshierba inicial del cultivo de maíz, contribuyendo a la mejora del bienestar de las personas dedicadas a esta labor.

1.4 Hipótesis

El diseño y construcción de un prototipo robotizado para la deshierba automática de cultivos de maíz en su etapa inicial para la Asociación Unión y Progreso del cantón Píllaro, provincia de Tungurahua, permitirá una eliminación de al menos el 70% de malas hierbas en el cultivo

1.5 Objetivos

1.5.1 Objetivo General

Desarrollar y construir un prototipo robotizado para el deshierbe automático en zonas planas de cultivos de maíz en su etapa inicial, a través de técnicas de visión artificial para la Asociación “Unión y Progreso” del Cantón Píllaro, Provincia de Tungurahua.

1.5.2 Objetivos Específicos

- Diseñar la plataforma móvil del prototipo robotizado de deshierba utilizando materiales y métodos óptimos para atravesar el cultivo de maíz.
- Desarrollar un sistema a través de técnicas de Visión Artificial que permita la navegación de la plataforma en hileras de cultivo y el reconocimiento de malas hierbas.
- Desarrollar un sistema de control para la navegación de la plataforma en hileras de cultivo y para el accionamiento de la herramienta de deshierbe del Robot Delta.
- Diseñar una herramienta eficaz para el Robot Delta encargada de remover la mala hierba en los cultivos de maíz.
- Lograr un resultado final de eliminación de mala hierba de al menos un 70% en el área trabajada por el robot

1.6 Alcance

El prototipo del presente proyecto dispondrá de:

- Estructura mecánica
- Manipulador de 3 grados de libertad
- Sistema de Accionamientos
- Sistema de Visión Artificial
- Efecto final

El alcance técnico del proyecto se enfoca en diseñar y construir una plataforma móvil, la cual transporte un manipulador robótico de 3 grados de libertad que alberga un efector final o herramienta que sea capaz de cortar y eliminar la maleza en campos planos de maíz en su estado inicial, que será controlado mediante técnicas de visión artificial y la aplicación de reconocimiento de patrones aplicando clasificadores de objetos.

CAPÍTULO II

MARCO TEÓRICO

2.1 Cultivo de Maíz

2.1.1 Maíz

El maíz es un cereal originario de América, se estima que tiene 7000 años de antigüedad desde su domesticación y hoy en día está entre los alimentos más importantes del mundo, su cultivo es global con mayor concentración en Europa y América del Norte. Es un cultivo muy importante para obtención de derivados como almidón, aceite, etanol, proteínas, entre otros derivados (FAO, 2008). Es una planta anual que crece en climas diversos, crece una planta por semilla normal, el maíz pueda alcanzar una altura de 2 – 2.5m, incluso hay casos excepcionales que la planta alcanza unos 5m. Su fruto es la mazorca, también conocida como *choclo* y en el Ecuador hay muchas variedades con distintas aplicaciones dependiendo de la temporada.

2.1.2 Desarrollo vegetativo del Maíz

El maíz posee un desarrollo dividido en siete etapas principales, sin embargo, se hará un énfasis en el desarrollo vegetativo, puesto que ésta es la más crítica para el correcto desarrollo de la planta.



Figura 3 Etapas del desarrollo del cultivo de maíz

Fuente: (SINAGAP, 2015)

En la etapa del desarrollo vegetativo, se puede subdividir en estados vegetativos en base al número de hojas de la planta, en donde según menciona (Fassio, Carriquiry, Tojo, & Romero, 1998), se las puede designar con la letra V, yendo desde V1 hasta V(n), donde “n” corresponde al número de la última hoja del maíz.

2.1.3 Siembra

Para la siembra de este cultivo, se recomienda según (SINAGAP, 2015) una distancia entre hileras de 75 a 80 cm y la distancia de entre plantas de 20cm si la siembra es de una sola semilla y 50 cm para más de una semilla.



Figura 4 Distribución del maíz en la Siembra

2.1.4 Deshierba del maíz

Uno de los principales problemas (especialmente en la etapa de desarrollo) que presentan los cultivos de maíz es la aparición de maleza o conocida comúnmente como malas hierbas, las cuales son plantas indeseables, generalmente silvestres ajenas al cultivo y que pueden aparecer en cualquier momento. El principal efecto que tienen es su competencia por los nutrientes, radiación solar y agua que se dan entre el cultivo y la maleza (AGROCALIDAD, 2014).

Es importante en un cultivo la eliminación las malas hierbas o maleza para promover el rápido y vigoroso desarrollo del cultivo (MAGAP, 2013), pues es

posible que ciertas especies de maleza alberguen insectos-plaga, nemátodos y otros organismos patógenos que atacan y causan enfermedades a las plantas (Cárdenas, 1987).



Figura 5 Cultivo de una leguminosa con alta presencia de maleza.

Fuente: (Pérez, Guillermo, & Fereira, 2012)

Es aconsejable llevar a cabo la escarda cuando el tiempo es cálido y seco, ya que esto hace más vulnerables a las malas hierbas. También un factor importante es el tiempo, debido a que en las etapas tempranas del cultivo, es cuando mayor daño puede causar la maleza. (Alejandro, 2012).

2.1.5 Tipos de Deshierba

a. Mecánico o Manual

Se trata de técnicas que no utilizan máquinas modernas ni productos químicos como herbicidas, limitándose a medios más naturales y tradicionales.



Figura 6 Deshierbe tipo manual de maíz en Píllaro

Debido a que en algunas zonas se dispone de mano de obra. Esta actividad comúnmente es de tipo familiar.

b. Deshierbe con herbicidas químicos

La industria de los agroquímicos ha permitido que el control químico sea hoy el medio de combate de malas hierbas más utilizado. Estos productos deben ser manipulados cuidadosamente, a fin de evitar accidentes que pueda comprometer la salud de los trabajadores y del medio ambiente (INIAP, 2003).



Figura 7 Deshierbe por herbicidas

Fuente: (Ruggerone, 2015)

2.1.6 Tipos de herramientas mecánicas de deshierba

En la actualidad hay tres tipos de técnicas físicas para el control de la mala hierba: (1) Enterramiento, (2) Cortadura, (3) Extirpado de la mala hierba. El enterramiento es realizado por herramientas de labranza que hacen más suave el suelo y entierran las malas hierbas para el control completo, mientras que las técnicas de cortadura y el extirpado se usan para la eliminación total de la mala hierba al arrancarla del suelo. La mayoría de las técnicas para la eliminación de maleza en cultivos de fase temprana, se realizan fuera de la hilera del cultivo (no entre plantas de la misma hilera) y unos pocos que son dentro de la hilera, los cuales tienen más cercanía con el cultivo sin afectarlo de ninguna forma. Según (Ahmad, Tang, & Steward, 2014) las más importantes son:

a. Desherbador de Canasta

Es un tipo de herramienta para el deshierbe que va acoplada a un tractor y no necesita de energía eléctrica para el movimiento de su 'canasta' porque el roce con el suelo permite que la herramienta gire sobre el mismo. Su principal función es eliminar la mala hierba de la parte superior del suelo y con velocidades entre 6.4 y 12.9 Km/h.

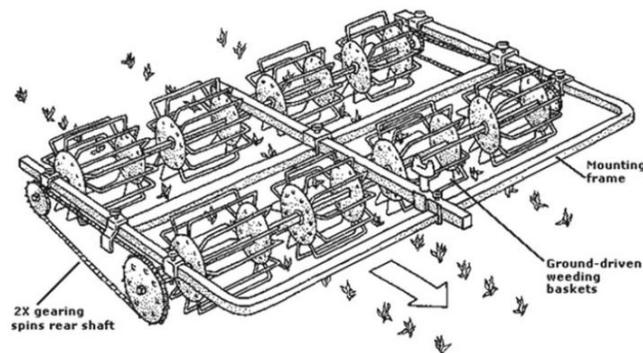


Figura 8 Herramienta de Deshierbe tipo Canasta

Fuente: (Ahmad, Tang, & Steward, 2014)

b. Desherbador de Dedos

Utiliza puntas de caucho que están sujetas a un disco en forma de cono de acero y distribuido a cierto ángulo. Los 'dedos' penetran el suelo justo debajo de la superficie para remover pequeñas malas hierbas.



Figura 9 Aplicación del Deshierbe de dedos

Fuente: (Ahmad, Tang, & Steward, 2014)

Esta desherbador puede ser usado en terrenos blandos y duros; la forma de la herramienta es excelente para malas hierbas jóvenes con una velocidad de operación recomendada de 4.8 a 9.7 Km/h. Según un estudio del 2004 se muestra que es eficiente en un 61%.

c. Desherbador de Torsión

Utiliza un marco rígido que tiene puntas de resortes conectados y encorvados, estas dos puntas están paralelas al suelo y se encuentran cerca de la hilera del cultivo. Este tipo de herramienta es capaz de reducir entre el 60-80% de las malas hierbas. Este tipo de desherbado requiere una baja velocidad de operación.



Figura 10 Desherbado con la Herramienta de Torsión

Fuente: (Ahmad, Tang, & Steward, 2014)

d. Desherbador de Cepillo

Son herramientas flexibles que están hechas de fibra de vidrio o nylon y rotadas alrededor de los ejes horizontal y vertical. Los cepillos son usados a través de potencia hidráulica del tractor. Esta herramienta principalmente extrae las malas hierbas, pero también las entierra y las rompe. Tiene una cubierta protectora para proteger el cultivo. Es usada para malas hierbas en las primeras etapas y tiene una eficiencia del 45 – 90% a una profundidad de 0,6 pulgadas. En

rangos de velocidad de 1.2 millas/h y a velocidad angular de 240 – 360 rpm es más efectiva que el deshierbe manual.



Figura 11 Herramienta rotatoria de deshierbo por cepillos

Fuente: (Ahmad, Tang, & Steward, 2014)

e. Eco-weeder

Es muy similar a la herramienta por cepillos, pero su principal diferencia es que es mecánicamente accionada por un operador humano y no requiere de potencia hidráulica.



Figura 12 Eco-weeder con Manipulación Humana y Accionamiento Manual

Fuente: (Ahmad, Tang, & Steward, 2014)

Recomendado para pequeños cultivos por su bajo costo de adquisición y mantenimiento. Se usa a una velocidad de 0.5 – 1.5 milla/h y la velocidad de la

herramienta (correa de transmisión) es de 150 – 300 rpm. Puede ahorrar en los costos de deshierbe manual hasta en un 60% y su efectividad no está probada todavía.

2.1.7 Tecnología automática en la eliminación de malas hierbas

Permite el reconocimiento de la hilera de cultivo, plantas benéficas y perjudiciales para que después remueva las malas hierbas a través de diversos métodos de eliminación. Estas herramientas son capaces de:

- Guiarse en la hilera
- Detección e identificación de mala hierba
- Preciso control de malas hierbas dentro de la hilera.
- Mapeo

El sistema de visión artificial de estos prototipos es capaz de reconocer con un bajo error a velocidades de 2.6 – 10 Km/h, el GPS puede permitir hasta un error lateral RMS de 2.4 pulgadas y una distancia de error máxima de 13 cm. Sin embargo, este tipo de sistemas necesita una siembra precisa, de ser posible utilizando tecnología GPS o similar. Los sistemas de identificación de malas hierbas usados en la actualidad son:

- Morfología biológica
- Características espectrales
- Estructura visual
- Algoritmo de segmentación ambiental adaptativa al entorno
- Segmentación de imágenes a color usando un algoritmo genético

2.2 Componentes de Robots

2.2.1 Actuadores eléctricos

Los actuadores eléctricos son transductores que transforman energía eléctrica en energía mecánica, para proporcionar movimiento a una máquina (Sánchez, 2008). Según el tipo de corriente y el tipo de funcionamiento utilizado se

pueden clasificar en motores de corriente continua (DC), motores de corriente alterna AC, motores paso a paso y servomotores, de los cuales, para el presente documento solo serán objeto de estudio los motores de corriente continua y servomotores.

a. Motores DC

Es uno de los motores más utilizados en el campo de la robótica, debido a su disponibilidad y por su fácil control. Además, son preferidos en aplicaciones donde se necesite usar baterías.

Un motor DC convencional consta de dos secciones principales, una que se encuentra fija (usualmente llamado estator) y otra la cual tiene la función de girar (rotor o armadura). (Cook, 2015)

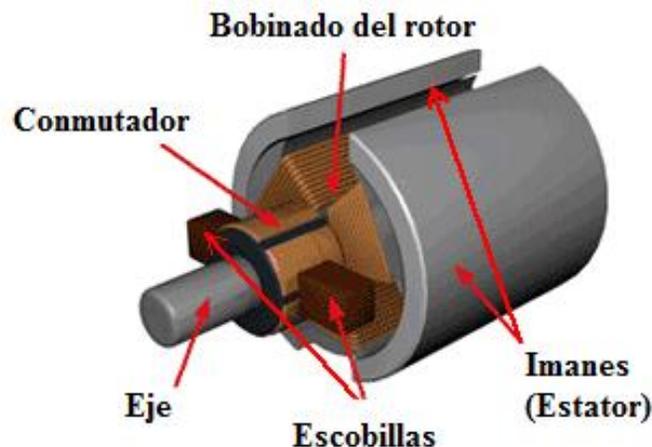
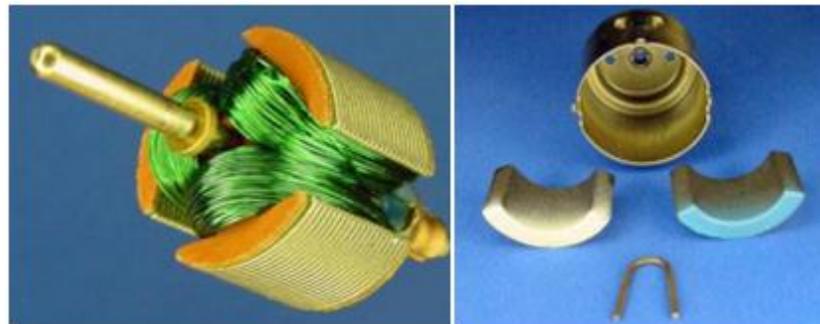


Figura 13 Partes de un motor DC convencional

Fuente: (Crespo, 2016)

En el rotor consta de devanados de alambre montados sobre un cilindro de material magnético, este se ubica dentro del estator, el cual tiene la función de generar un campo magnético que puede ser a través de imanes permanentes o electroimanes (Bolton, 2011).



(a) Rotor

(b) Estator

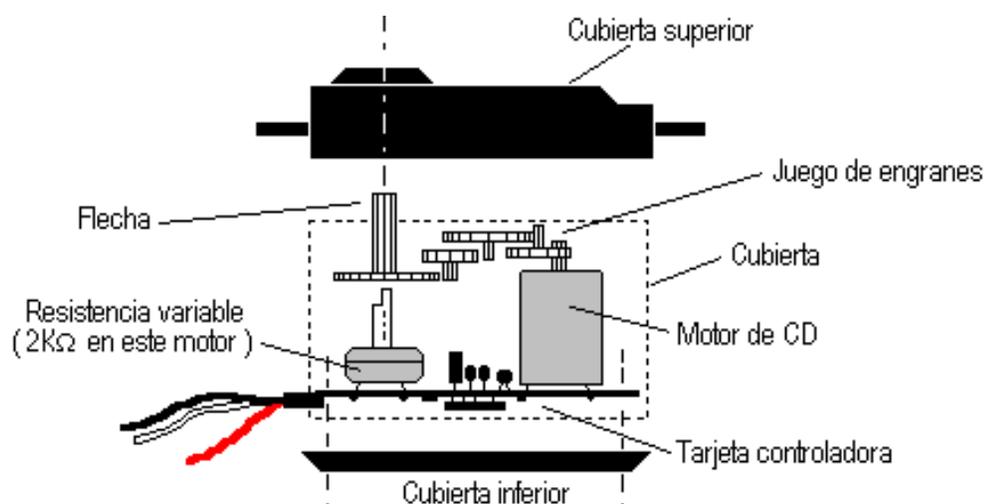
Figura 14 Rotor y Estator en un motor DC

Fuente: (Cook, 2015)

Los extremos de los devanados de la armadura se conectan a los segmentos adyacentes de un anillo llamado conmutador y la relación eléctrica se logra mediante contactos de carbón llamados escobillas. (Bolton, 2011)

b. Servomotores

Son un tipo especial de motores DC, la principal característica es que su eje posee la capacidad de ubicarse en un ángulo específico, consta de un motor DC, un circuito de control, una caja reductora, circuito de alimentación. (Cook, 2015)

**Figura 15 Vista frontal y lateral interna de un servomotor**

Fuente: (Cook, 2015)

El control del posicionamiento del eje del servomotor se lo realiza mediante un tren de pulsos, el cual debe tener una frecuencia específica dependiendo de la marca, y al variar la duración de los pulsos, se puede conseguir el ángulo deseado.

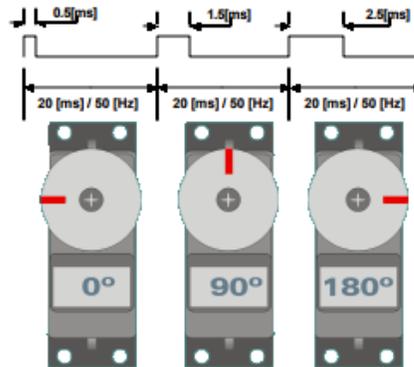


Figura 16 Tren de pulsos para el control del ángulo de un servomotor

Fuente: (Cirelli, 2015)

2.2.2 Ruedas

a. Ruedas Estándar

Este tipo de rueda tiene dos grados de libertad (rotación motorizada a través del eje de la rueda y del punto de contacto) además el ángulo entre el chasis del robot y la rueda es constante. Posee la característica de que su movimiento de dirección no tiene un efecto secundario (Siegwart, Nourbakhsh, & Scaramuzza, 2011).



Figura 17 Rueda de Carretilla

Fuente: (SODIMAC, 2017)

b. Rueda Loca

La rueda loca (también conocida como castor) es una rueda sin tracción, simple o doble, que puede girar libremente y que generalmente está situada en la parte inferior de una estructura, puede ser de dos tipos rígida y giratoria, de las cuales la segunda permite girar libremente en cualquier dirección. (Electric Bricks, 2010). Son muy utilizadas en robótica móvil, en especial la rueda loca giratoria en la configuración diferencial, ya que proporciona al robot una buena orientación y evitar cabeceos.



(a) Rígida

(b) Giratoria

Figura 18 Rueda loca

Fuente: (Electric Bricks, 2010)

2.2.3 Sensores

Los sensores son parte muy fundamental para muchas áreas de la Ingeniería, pues forman parte de muchos sistemas complejos como en la Mecatrónica y Robótica, se puede hacer analogía a los sensores como los sentidos del cuerpo humano, pues permite a los sistemas percibir las variables que se encuentran a su alrededor, los sensores actuales tienen multitud de principios de funcionamiento y pueden medir una multitud de variables entre las cuales se pueden medir posición, ángulo, temperatura, fuerza, contacto, proximidad, etc. Los sensores de un robot se pueden catalogar en dos categorías, como se menciona a continuación (Universidad de Valladolid, 2005):

a. Sensores Internos

Sensores integrados en la propia estructura mecánica del robot, estos sensores proveen información del estado del robot: fundamentalmente de la posición, velocidad y aceleración de las articulaciones.

La información que se debe obtener para poder controlar un robot a través de sus sensores internos son: posición, velocidad y aceleración. Para medir estas variables, se pueden usar los siguientes elementos:

- **Potenciómetros**

Se usan para la determinación de desplazamientos lineales o angulares. Esta diferencia de voltaje puede medirse y disponer de un sistema para el calibrado de manera que por cada potencial se obtenga proporcionalmente una distancia de desplazamiento.

- **Encoders**

Constan de un disco ranurado o marcado con una serie de marcas opacas colocadas radialmente y equidistantes entre sí, además de un sistema de iluminación y de un elemento fotorreceptor.

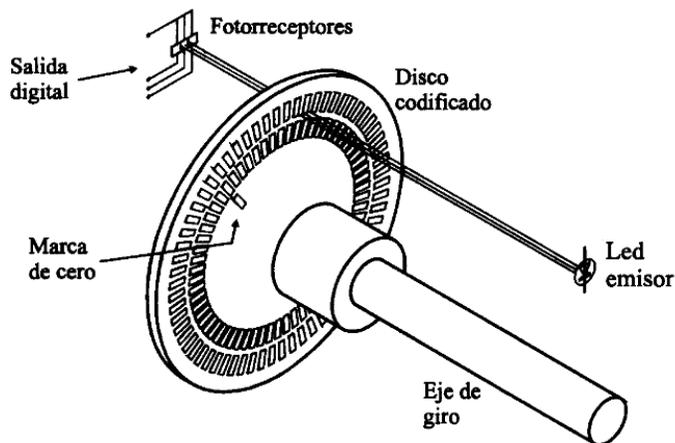


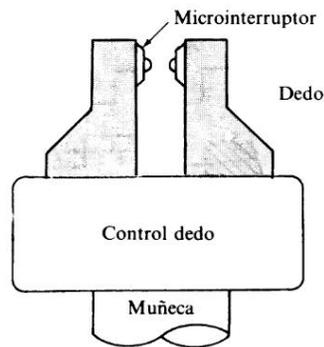
Figura 19 Diagrama típico de un encoder

Fuente: (Universidad de Valladolid, 2005)

Se debe acoplar un eje al disco para medir su posición, a medida que el eje gira se van generando pulsos en el receptor cada vez que la luz atraviese las marcas (o rebote, dependiendo del tipo de sensor que recoja las marcas), con el dato de la cuenta de estos pulsos es posible conocer la posición del eje. La resolución es directamente proporcional al número de marcas que se pueden poner físicamente en el disco.

b. Sensores Externos

Dan información del entorno del robot: alcance, proximidad, fuerza, contacto, etc. Sus aplicaciones se dan para, para identificación y manipulación de objetos y guiado de robots



(a) De contacto



(b) Ultrasónico

Figura 20 Sensores externos

Fuente: (Universidad de Valladolid, 2005)

2.2.4 Sistemas de alimentación

Un robot, de acuerdo a su aplicación, puede ser de tipo fijo o de tipo móvil. Pero en cualquiera de los dos casos se necesita de un tipo de alimentación que debido a las condiciones de los elementos internos de un robot, es de tipo eléctrica. La alimentación debe proporcionar la energía necesaria para todo el sistema (MEC, 2003) y debe ser suficiente para que el robot al menos cumpla con un ciclo completo de trabajo, un robot puede adquirir dos tipos de energía eléctrica, como se presenta a continuación:

a. Corriente Alterna

Es un tipo de energía que se manifiesta en períodos iguales de tiempo, tiene la ventaja de fácil transmisión y alta potencia, sin embargo, necesita de personal especializado para su manipulación, este tipo de energía es preferida cuando el trabajo es fijo y que necesite de un flujo constante de energía.

b. Corriente Continua

Modelo preferido para robots móviles debido a que puede ser almacenada para su transporte, este tipo de energía puede ser aprovechada a través de un conversor AC/DC o a través de la generación por paneles solares, su almacenamiento se da en baterías recargables como las de composición de Plomo y Ácido, que proveen de energía a los robots móviles.



Figura 21 Batería de Plomo y Ácido a 12V y 20Ah

Fuente: (zBattery, 2017)

2.2.5 Microcontroladores

Los microcontroladores son una parte fundamental en el avance de las ciencias de la Electrónica, su desarrollo ha sido parte una parte crucial para el desarrollo de sistemas actuales, los microcontroladores son pequeños elementos autónomos que contienen un procesador, una memoria y periféricos además pueden ser usados como sistemas embebidos (Future Electronics, 2017). Sus aplicaciones son muy variadas, contando en las principales como sistemas

autónomos, herramientas de potencia, juguetes, equipo médico, máquinas de oficina y otras aplicaciones embebidas como los controladores Arduino.



Figura 22 Microcontrolador PIC16F628A

Fuente: (Future Electronics, 2017)

a. Características Microcontroladores

Entre las características más importantes para su uso, se pueden mencionar las siguientes (Ecured, 2017):

- Reducen el costo económico y consumo de energía en sistemas complejos.
- Sus especificaciones se adaptan de acuerdo a la complejidad de la aplicación.
- Relativamente fáciles de mantener y reemplazar.
- Modularidad.
- Variedad de lenguajes de programación para su asignación de sus funciones.
- Flexibilidad de uso.
- Protocolos de comunicación estándar como Serial, SPI, I2C, etc.

b. Sistemas Embebidos

A estos sistemas se les conoce como sistemas computacionales diseñados para realizar una o varias funciones dedicadas, frecuentemente en un sistema de computación en tiempo real (UNED, 2011). Su aplicación es muy variada, debido a que se pueden ocupar en tarjetas de video, procesadores digitales de señales (DSP), módem, audios, etc. Comúnmente vienen incluidos en los PCB de productos comerciales. Al igual que los microcontroladores, una de su ventaja es su tamaño y entre algunos ejemplos de estos se pueden mencionar: Raspberry Pi,

BeagleBone Black, Nvidia Tegra, etc. Entre las principales características que poseen estos sistemas están (UNED, 2011):

- Confiabilidad para que el sistema trabaje a lo largo del tiempo.
- Mantenibilidad de los componentes, para trabajar luego de un fallo.
- Disponibilidad de repuestos para mantenimiento.
- Seguridad informática.
- Eficientes en consumo de energía, tamaño, peso, costo.
- Interfaces de usuario dedicadas (sin ratón, teclado o pantalla)

2.2.6 Sistemas de suspensión

La suspensión es una parte muy importante en sistemas móviles, debido a que ayudan a afrontar cualquier irregularidad en la superficie de desplazamiento, en el caso del sector automovilístico proporciona confort a los ocupantes, así como seguridad a la carga que se transporta al disminuir las vibraciones. Además, uno de sus principales objetivos es asegurar el contacto de la rueda con el suelo, esto favorece notablemente en el control, estabilidad y tracción del sistema móvil. (Luque, Álvarez, & Vera, 2004)

Un sistema de suspensión se ubica como la unión entre la estructura y la rueda, y consta de dos tipos de elementos: los elásticos (muelles helicoidales o ballestas) y los de disipación (amortiguadores).



(a) Resortes helicoidales y Ballestas¹ b) Amortiguadores²

Figura 23 Partes de un sistema de suspensión

Fuente:¹ (Corporación Supertren, 2017), ² (Taller Autojesa, 2017)

Los elementos elásticos tienen la función de proporcionar un adecuado contacto de la rueda con el piso, además de irse adaptando a las irregularidades que se puedan presentar en la superficie, mientras que los elementos de disipación o amortiguadores neutralizan oscilaciones provocadas por los elementos elásticos (Rocha, 2013).

2.2.7 Unidad de Control

La unidad de control está conformada por los elementos computacionales y el *software* que regulan el comportamiento global del robot (MEC, 2003). Es una de los elementos más importantes para que un robot funcione bien, pues permite realizar rutinas de acuerdo a los elementos físicos con los que esté compuesto el robot. Para poder controlar un robot, se pueden presentar los siguientes controles básicos (Rice University, 2017):

a. Control de realimentación

Permite que el robot desarrolle su objetivo de control a través del monitoreo y corrección de su variable de interés, para lograr su función.

b. Control de lazo abierto

Es un tipo de control donde no hay realimentación de la acción comandada y por lo tanto no hay corrección en el estado del sistema, funciona bien cuando los sistemas son muy predecibles aunque sin un rendimiento alto.

c. Control de realimentación hacia adelante

Usa una señal comandada como una función de algunos parámetros medidos con anterioridad, este tipo de método no se basa en el resultado real de un comando de movimiento, sino que se hace un cálculo para que el control sea más preciso.

2.3 Robótica Móvil

Se puede considerar como una rama de la robótica, que se enfoca en el desarrollo del diseño de hardware y software de robots móviles, su principal meta es la búsqueda de una buena funcionalidad y robustez (Nehmzow, 2006). Actualmente la robótica móvil también se enfoca en hallar nuevos usos para los robots, así como mejorar su autonomía. En el diseño mecánico de estos, se pueden presentar distintas formas de locomoción o traslación (Ollero, 2001), como se presenta a continuación:

2.3.1 Vehículos con ruedas

Es una de las formas de movilidad que posee una gran eficiencia y es una solución simple para terrenos duros y libres de obstáculos. Además, proporciona altas velocidades, sin embargo, entre sus desventajas están que se puede presentar deslizamiento y es susceptible a ser afectado por las vibraciones. Entre las configuraciones más importantes de este tipo de robots están:

a. Ackerman

Esta configuración se presenta en vehículos comunes de cuatro ruedas, donde el valor de los ángulos de giro de las ruedas es muy cercano, pero nunca igual, consiguiendo así evitar o minimizar el deslizamiento en las ruedas (Ollero, 2001).

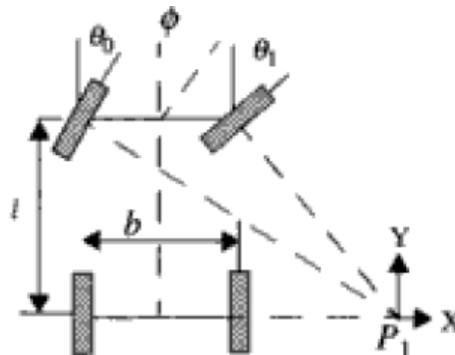


Figura 24 Sistema Ackerman

Fuente: (Ollero, 2001)

b. Triciclo Clásico

Es un sistema tradicional y bastante común, su principal característica es su simplicidad, la rueda delantera tiene la función proporcionar tracción y direccionamiento al mecanismo. Su maniobrabilidad es mayor que la Ackerman, pero presenta problemas de estabilidad en terrenos difíciles debido a que en ellos, su centro de gravedad puede variar, lo que genera una tracción menor a la normal.

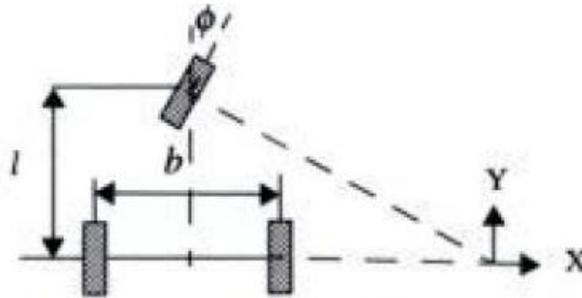


Figura 25 Triciclo Clásico

Fuente: (Ollero, 2001)

c. Direccionamiento Diferencial

Está caracterizado porque las velocidades de sus ruedas laterales no son iguales, la tracción se consigue también con las mismas ruedas. Adicionalmente existen una o más ruedas de soporte, esta configuración es la más frecuente para ambientes internos.

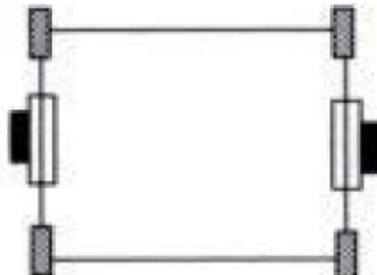


Figura 26 Sistema de direccionamiento diferencial

Fuente: (Ollero, 2001)

d. Otros

Existen otras configuraciones poco comunes, pero no menos importantes, como la configuración Skid Steer que en cada lado tiene ruedas que se mueven al mismo tiempo, obteniendo así una combinación entre las velocidades de las ruedas izquierda y derecha. Además, existe la configuración Pistas de Deslizamiento que contiene orugas y la impulsión se consigue mediante las mencionadas pistas de deslizamiento.



(a) Skid Steer

(b) Pistas de deslizamiento

Figura 27 Otras configuraciones

Fuente: (Ollero, 2001)

2.3.2 Navegación de Robots Móviles

La navegación es la disciplina que proporciona a un robot móvil la capacidad de guiar su curso a través de un entorno con presencia de obstáculos, de forma segura, permitiéndole llegar a salvo a su sitio objetivo.

Según (Muñoz, 1997) existen tres tareas consecutivas que se deben llevar a cabo en la navegación: la percepción del entorno, la planificación de una trayectoria libre de obstáculos y el guiado del vehículo a través de la trayectoria definida, existen algunas técnicas de navegación usadas por los robots, entre ellas se analizará la odometría.

2.3.3 Localización de Robots Móviles

La localización de los robots móviles permite el seguimiento de precisión de las trayectorias definidas, para que un sistema pueda seguir una trayectoria es necesario que el robot móvil disponga de una definición precisa de donde está posicionado su sistema de referencia. La localización se da gracias a sensores a bordo que presentan información incremental de los movimientos del robot. Hay algunas técnicas que se puede aplicar para la localización de robots móviles entre ellas se encuentran *dead-reckoning*, sistemas de visión, incorporación de filtros de Kalman y utilización de dispositivos de posicionamiento global (GPS).

2.3.4 Dead Reckoning

El dead reckoning (navegación por estima) es un método sencillo, robusto y de bajo costo usado para la navegación de robots móviles, su uso se remonta a la antigüedad para la navegación en mares, el dead reckoning es un método para determinar (sin la ayuda de observaciones externas) la posición y orientación de un vehículo a partir del registro de los cursos recorridos, la distancia tomada y la deriva estima conocida (Roston & Krotkov, 1991). Para su implementación más básica en robots móviles es necesario que el sistema cuente con un odómetro que habitualmente es un encoder óptico.

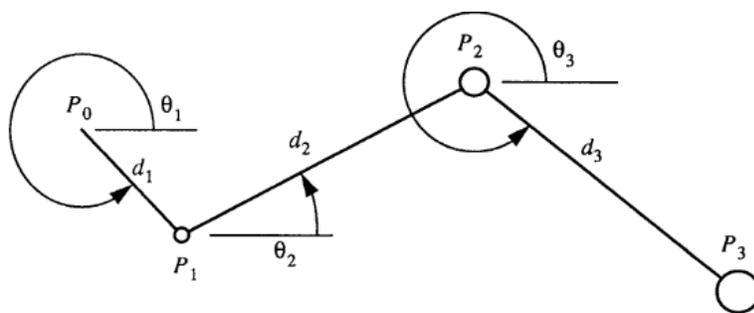


Figura 28 Trayecto establecido por dead reckoning

Fuente: (Roston & Krotkov, 1991)

a. Odometría

Es una de las técnicas utilizadas en la navegación de robots móviles, donde se puede estimar la posición y la orientación mediante la utilización de sensores, que generalmente miden la revolución de las ruedas (Moreno, Blanco, & González, 2007).

Existen diferentes formas de aplicar esta técnica de acuerdo al tipo de configuración del robot móvil de estudio, en donde se tienen fórmulas matemáticas para cada una, para lo cual se requiere conocer algunos parámetros del robot como: distancias entre las ruedas, radio de la rueda, resolución del sensor, entre otros.

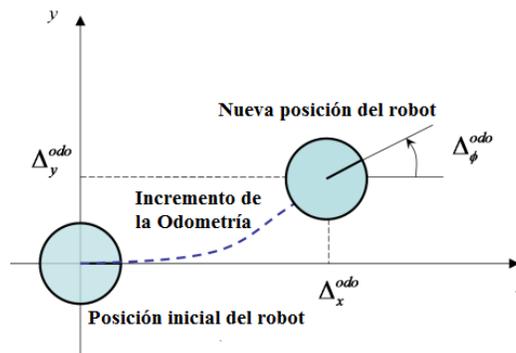


Figura 29 Odometría de un robot móvil

Fuente: (Blanco, 2013)

2.4 Robots Paralelos

2.4.1 Definición

Este tipo de robots se constituyen de cadenas cinemáticas cerradas, de tipo seriales, con las cuales su base fija se une con su efector final o herramienta (Merlet, 2006).

Según (Aracil, Roque, Reinoso, & Sabarter, 2006), los robots paralelos presentan ciertas características beneficiosas, como la capacidad de levantar pesos mayores al suyo, alta eficiencia energética, ligereza, alta rigidez, precisión y

velocidad. Sin embargo, también presentan ciertos inconvenientes como su cinemática compleja y su limitado volumen de trabajo.

2.4.2 Robot tipo Delta

Este robot de tipo paralelo, está constituido por tres cadenas cinemáticas cerradas similares, dispone de una plataforma fija en donde se ubican los actuadores y una móvil en la cual se encuentra su efector final o herramienta, constituyéndose así un total de cuatro grados de libertad, uno de rotación y el resto de traslación (Liu, Wang, & Pritschow, 2005)

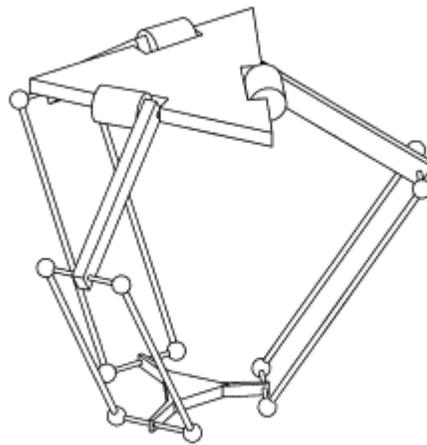


Figura 30 Robot Delta clásico

Fuente: (Liu, Wang, & Pritschow, 2005)

2.4.3 Configuraciones de los Robots tipo Delta

Según (Demaurex, 2013), se puede encontrar cuatro configuraciones habituales en los robots tipo Delta, entre las cuales se encuentran el Delta clásico, el Delta de tres ejes verticales, el de tres ejes paralelos y el de tres ejes competidores, siendo la principal característica de cada uno el espacio o volumen de trabajo que pueden describir y la forma en que se ubican los actuadores.

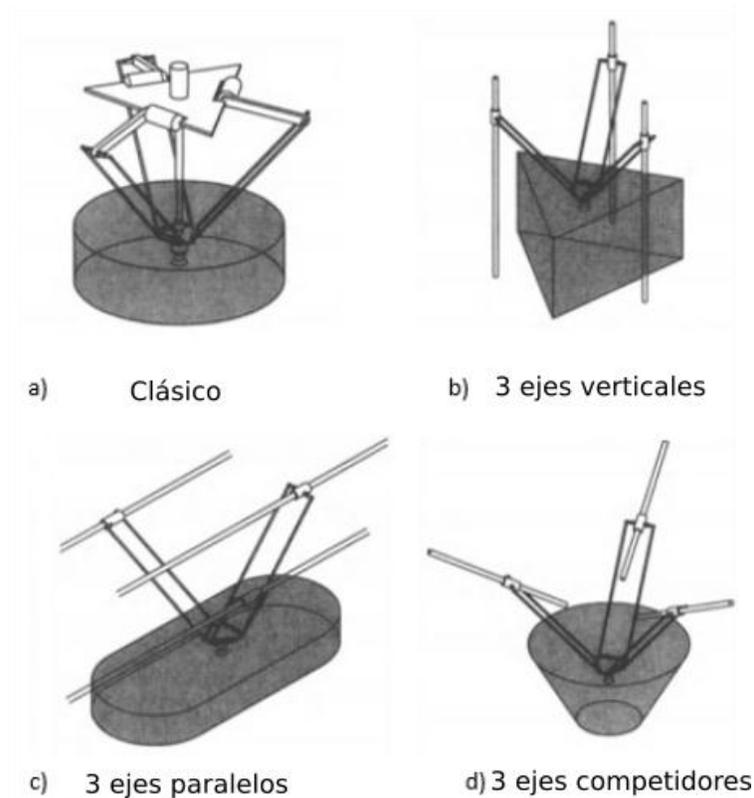


Figura 31 Configuraciones del robot Delta

Fuente: (Demaurex, 2013)

2.4.4 Cinemática del robot tipo Delta

La cinemática de un robot, busca establecer una relación entre las posiciones de las articulaciones y la posición del efector final, para lo cual generalmente hace uso de un análisis geométrico, debido a su baja complejidad y fácil programación de control en cualquier microcontrolador. Se divide en dos tipos:

a. Cinemática Directa

La solución para el este modelo consiste en calcular el valor de la posición de un punto $P(P_x, P_y, P_z)$ en el que está ubicada la herramienta, teniendo como dato el valor de los ángulos de las articulaciones del robot θ_{1i} (Salazar & Pachacama, 2006).

b. Cinemática Inversa

La solución del modelo cinemático inverso permite encontrar los ángulos de entrada a las articulaciones del robot θ_{1i} para una posición de la herramienta en un punto $P(P_x, P_y, P_z)$ (Salazar & Pachacama, 2006).

2.5 Visión Artificial

Para los seres humanos, una de las herramientas de vital importancia para la realización de las actividades diarias, es el sentido de la vista. Con la ayuda de este sentido es posible percibir nuestro entorno, detectar formas, distancias, colores, luminosidad, etc. En un esfuerzo por entender cómo funciona nuestra visión, se han realizado muchas investigaciones al respecto, las cuales han permitido el desarrollo de herramientas matemáticas para poder expresar tridimensionalmente la forma de los objetos y poder realizar operaciones sobre las mismas (Szeliski, 2010).

Todas estas herramientas, potenciada con los crecientes avances informáticos que se ha venido dando, han dado paso la rama conocida como Visión Artificial. Esta busca imitar y plasmar el funcionamiento del sistema de visión humano, en máquinas o robots y de esta manera les permita cumplir sus funciones, para lo cual se utilizan imágenes digitales.

2.5.1 Imagen Digital

Una imagen digital puede ser considerada como una representación discreta de datos que poseen tanto información espacial (disposición) como intensidad (color) (Solomon & Toby, 2011). Se la puede representar mediante una matriz o arreglo bi-dimensional de valores numéricos, donde cada celda representa un pixel, (Call, Malespina, Varela, & Palomeque, 2007) el cual según manifiesta (Montabone, 2010), es la unidad más pequeña que se puede medir en una imagen y puede representar una magnitud física, que puede ser: la cantidad de luz, valor de un color, nivel de radiación infrarroja, profundidad, entre otras.

Dependiendo el tipo de imagen también se puede saber la cantidad de información que puede tener un pixel. Por ejemplo, en el caso de una imagen a color (RGB), el pixel tendrá 3 valores correspondientes a cada capa de color, en cambio si se trata de una imagen en escala de grises, el pixel representará un solo valor entre 0 y 255.

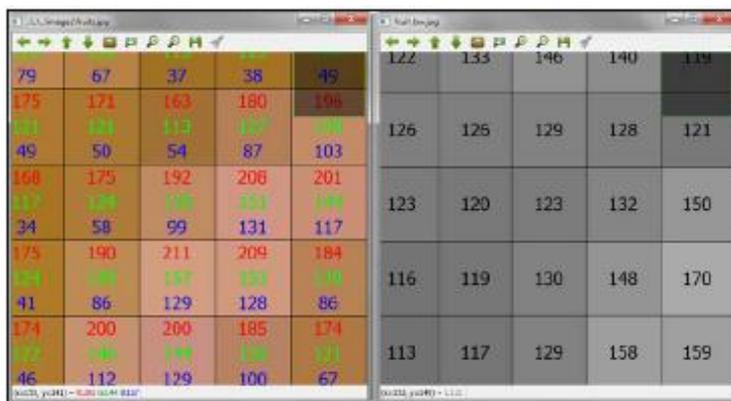


Figura 32 Pixeles en una imagen RGB y en escala de grises

Fuente: (Bueno, et al., 2015)

2.5.2 Etapas de un sistema de Visión Artificial

Según argumenta (Nuñez, 2016), todo sistema de Visión Artificial por lo general cumple cuatro actividades principales, las cuales son: captura, pre-procesamiento, segmentación y reconocimiento.

a. Captura

Se refiere a la obtención de imágenes digitales, la cual se lleva a cabo mediante la utilización un sensor óptico o cámara. (Nuñez, 2016). En cuanto a los sensores de las cámaras digitales, según menciona (Szeliski, 2010) existen dos tipos de sensores, los CCD (dispositivo de carga acoplada) y CMOS (semiconductor complementario de óxido metálico), la cual es la más utilizada hoy en día debido a que ofrece mejores características que la CCD.

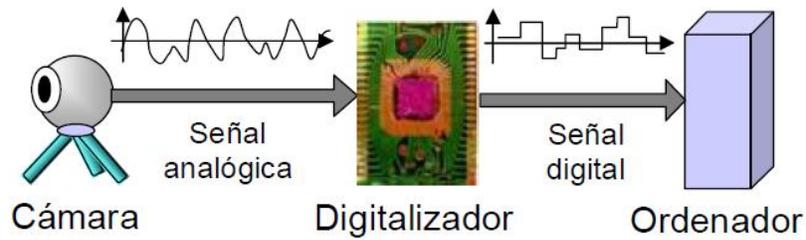


Figura 33 Proceso de captura de una Imagen Digital

Fuente: (Call, Malespina, Varela, & Palomeque, 2007)

En esta etapa se define algunas características que adoptará la imagen, como lo son su resolución y tamaño, las cuales dependerán del sensor, y algunos programas para procesamiento de imágenes permiten modificarlos de acuerdo a las necesidades de la aplicación.

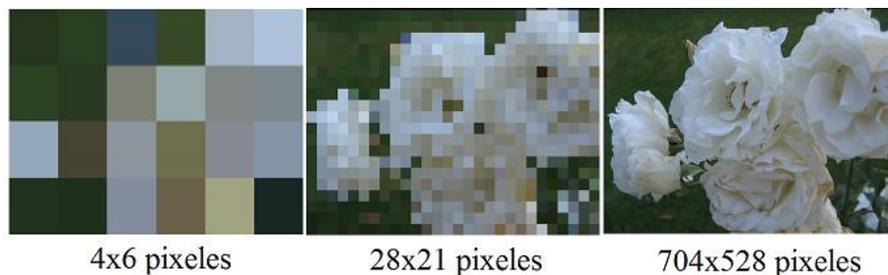


Figura 34 Resolución en la representación de una imagen

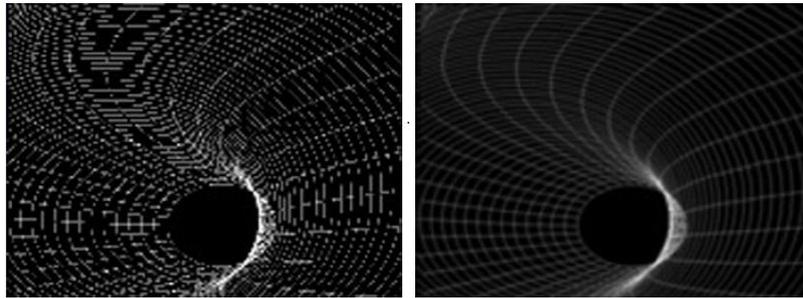
Fuente: (Montabone, 2010)

b. Pre-procesamiento

En esta etapa se utiliza herramientas matemáticas que permitan eliminar información innecesaria y mejorar información que es de interés para el usuario. Para esto se puede utilizar filtros, ajuste de brillo, contraste y otras técnicas que pueden ser representadas mediante transformaciones matemáticas.

El filtrado es un proceso para modificar o mejorar imágenes, enfatizando ciertas características o quitando otras. El filtrado de imágenes determina el valor de salida de un determinado píxel situado en una posición (x, y) realizando

algunas operaciones con los valores de los píxeles que se encuentran rodeando al mismo y generalmente son llamados píxeles vecinos (Bueno, y otros, 2015).



(a) Imagen Original

(b) Imagen Filtrada

Figura 35 Ejemplo de la utilización de un filtro

Fuente: (Brahmbhatt, 2013)

c. Segmentación

La segmentación de imágenes consiste en encontrar y separar un conjunto o agrupación de píxeles, los mismos que tienden a compartir una o más características como forma o color (Szeliski, 2010).

Entre las formas más utilizadas para segmentar, se puede mencionar la segmentación por color, por textura, por formas y por movimiento. Una de las técnicas más útiles, es la separación de colores por canales, especialmente útil cuando el objeto a segmentar corresponde a un canal en un espacio de color (como el RGB), por lo que se puede obtener su componente individual al restar de éste canal los demás canales de la imagen.

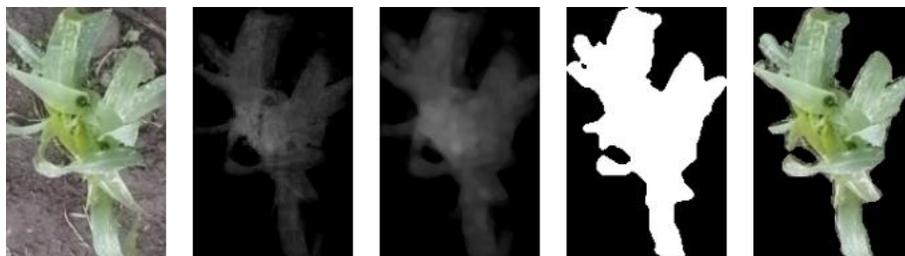


Figura 36 Imagen de una planta, con segmentación canal verde

d. Reconocimiento

El reconocimiento o clasificación es la última fase en un sistema de visión artificial y es con la cual un sistema puede tomar decisiones y actuar en base a las mismas. Existe algunas técnicas para poder reconocer imágenes procesadas entre ellas se pueden mencionar: Redes Neuronales Artificiales(ANN), Máquinas de Vectores de Soporte(SVM) o la más importante en la actualidad que es el Aprendizaje Profundo(Deep Learning.).

2.6 Aprendizaje Profundo (Deep Learning)

2.6.1 Neurona Artificial

Según (Lara, 2005), una neurona artificial o nodo, se considera como una unidad básica de procesamiento. Es un modelo matemático, el cual busca realizar funciones similares a la de una neurona biológica y se lo puede representar de la siguiente manera:

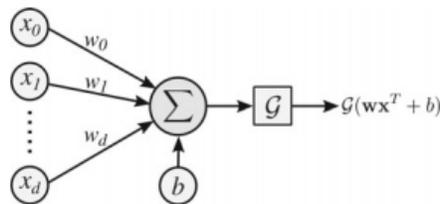


Figura 37 Representación simplificada de una neurona artificial

Fuente: (Lara, 2005)

En una neurona artificial las entradas son multiplicadas cada una por un valor llamado peso sináptico w_i , luego se suman entre si junto con un valor de ganancia o umbral llamado bias b , el resultado de dicha suma se evalúa en una función llamada función de activación, la cual genera un valor salida de la neurona activándola o desactivándola. Una neurona asimila su entorno y aprende, para ello modifica y adapta los valores tanto de los pesos sinápticos como del bias, a este proceso se le llama aprendizaje o entrenamiento (Tablada & Torres, 2007).

2.6.2 Redes Neuronales Artificiales

Las redes de neuronas artificiales o RNA, están formadas por un conjunto de neuronas altamente interconectadas entre sí. (Torres L. , 2010) Constan de tres grupos conformados por capas, entre las cuales están: la capa de entrada, una o varias capas ocultas y la capa de salida. Según el número de capas ocultas de puede definir el tipo de red, si hay pocas capas, la red se denotara como superficial, y si hay muchas capas como red profunda.

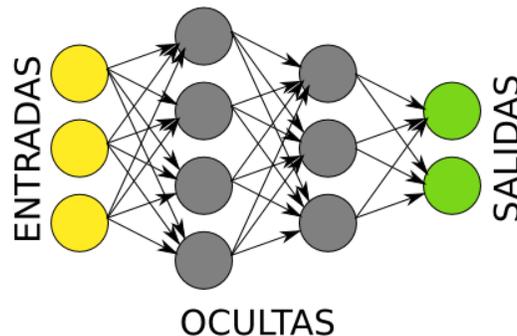


Figura 38 Red neuronal común de cuatro capas y sus partes constitutivas

Fuente: (López, 2014)

2.6.3 Aprendizaje de las Redes Neuronales

Entre los parámetros más importantes al momento de utilizar redes neuronales, está el de determinar el valor de los pesos sinápticos a utilizar entre las conexiones de las capas, para lo cual son utilizados dos métodos: propagación hacia atrás (backpropagation) y descenso o reducción del gradiente (Nuñez, 2016).

a. Backpropagation

Es un procedimiento iterativo que ajusta los parámetros de los pesos de la red de acuerdo con el gradiente de una medida o función de error. El procedimiento se implementa calculando un valor de error para cada unidad de

salida, y hace una retropropagación de los valores de error a través de la red (Sammut & Webb, 2017).

b. Reducción del Gradiente

El descenso de gradiente es un algoritmo de optimización para un sistema, cuya tarea es aproximar una función dada, su error de rendimiento se da en términos de los parámetros del sistema. El descenso del gradiente intenta minimizar este error ajustando los valores de los parámetros de forma incremental (Brinder, Hirokawa, & Windhorst, 2009).

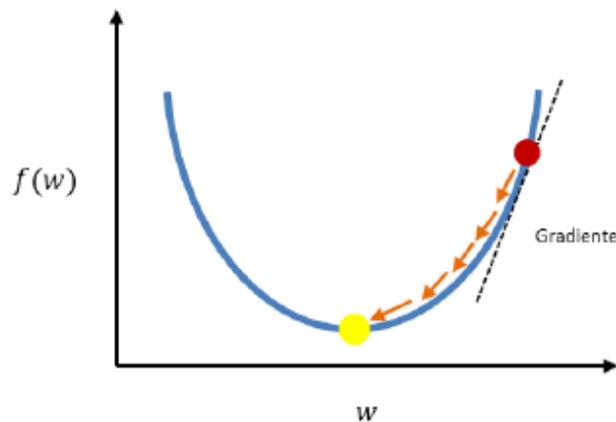


Figura 39 Descenso del gradiente

Fuente: (Nuñez, 2016)

c. Función de activación en Redes Neuronales

Existen algunos tipos de funciones que comúnmente son aplicadas sobre redes neuronales, dichas funciones deben tener la característica de ser no lineales. Esto permite que la red aprenda cualquier función no lineal siempre y cuando existan suficientes neuronas y capas en la red (Habibi & Jahani, 2017). Existen muchas funciones de activación para redes neuronales, a continuación, se muestran las más utilizadas.

Tabla 2
Funciones de activación para redes neuronales

Nombre	Gráfica Función	Gráfica de la Derivada
Sigmoide $G(x) = \frac{1}{1 + e^{-x}}$		
Tangente Hiperbólica $G(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$		
Softsign $G(x) = \frac{x}{1 + x }$		
Unidad Lineal Rectificada (ReLU) $G(x) = \max(0, x)$		

Fuente: (Habibi & Jahani, 2017)

Como se puede observar cada función tiene su derivada, esta es una condición necesaria de las funciones de activación, puesto que sin ella no es posible hacer uso del método de reducción o descenso del gradiente para reducir el error de entrenamiento. Las funciones sigmoide, tangente hiperbólica y softsign, son utilizadas para redes neuronales superficiales, debido a que, si se aplica estas funciones para redes profundas, en la ejecución del backpropagation el valor de su

gradiente se reduce hasta desvanecerse. Por esta razón para redes profundas la función mayormente utilizada es la Unidad Lineal Rectificada (ReLU) (Sammut & Webb, 2017).

2.6.4 Redes Neuronales Convolucionales

También son denominadas CNN por sus siglas en inglés, son una de las arquitecturas de aprendizaje profundo (Deep Learning) más utilizadas, y la aplicación más común que tienen son las relacionadas con visión por computadora y procesamiento de voz (Nuñez, 2016).

La arquitectura de este tipo de redes neuronales pueden variar, sin embargo en general constan de cuatro capas, una capa de entrada, una o varias capas convolucionales, una o varias capas de pooling, una por cada capa de convolución, y una capa totalmente conectada (FC o Full Connected) (Nuñez, 2016).

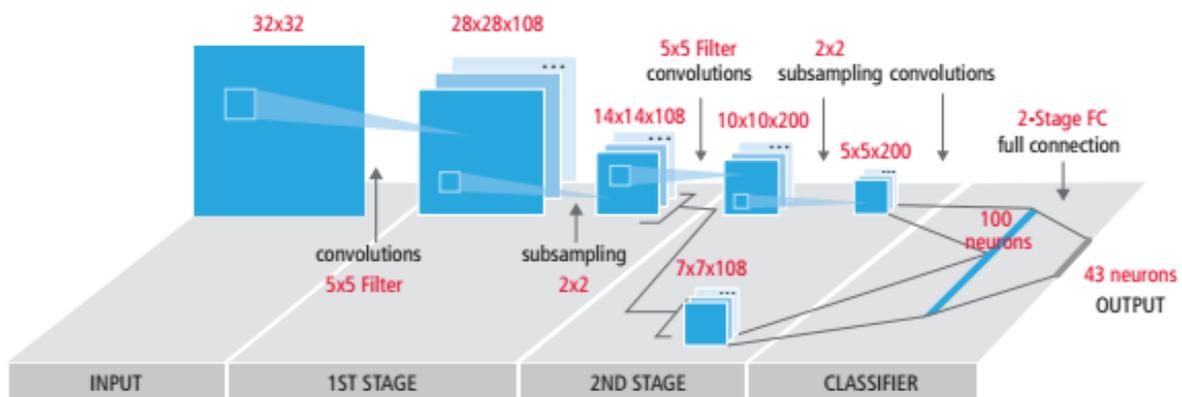


Figura 40 Diagrama típico de una Red Neuronal Convolutional

Fuente: (Hijazi, Kumar, & Rowen, 2015)

a. Capas Convolucionales

La convolución es la operación que hace posible la identificación y extracción de las características más significativas de la imagen de entrada como

esquinas, bordes, tonalidades, etc. Además permite que las neuronas compartan los pesos y bias, lo que reduce el número de parámetros de la red (Nuñez, 2016).

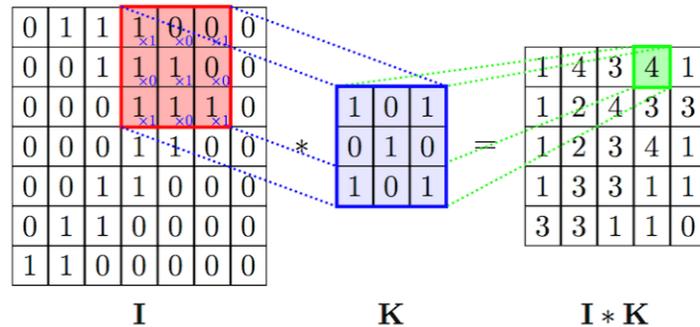


Figura 41 Operación de Convolución sobre una matriz

Fuente: (Veličković, 2017)

b. Capa de Pooling

Son las capas que sintetizan y reducen la información recogida, dejando solo los valores máximos en un área determinada de la capa de convolución. Esto hace que se pueda disminuir el volumen de información de entrada para la próxima capa (Nuñez, 2016).

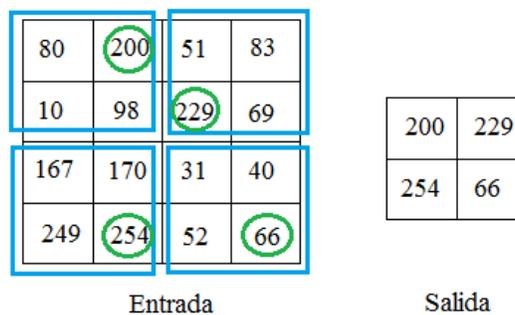


Figura 42 Función de Pooling o Subsampling, con valores al azar

c. Capa full connected (FC) o capa totalmente conectada

Generalmente es la última capa de la red neuronal, y es la que se encarga de calcular las puntuaciones obtenidas por una imagen que ingresó a la red,

según las clases que se tengan establecidas en la red. Posee muchas más conexiones que una capa convolucional y por ende un mayor peso.

2.6.5 Arquitecturas de Red

Las redes neuronales convolucionales son muy por tener muchas aplicaciones (Raszavian, Azizpour, Sullivan, & Carlsson, 2014), esto se debe a que hay múltiples arquitecturas que puedan realizar una función dependiendo del nivel de aprendizaje que se requiera, una de las aplicaciones más comunes en las CNN para agricultura de precisión es la clasificación de plantas benéficas de la maleza, para este caso, han demostrado tener un excelente porcentaje de clasificación la arquitectura *sNet* y *cNET* (Potena, Nardi, & Pretto, 2016).

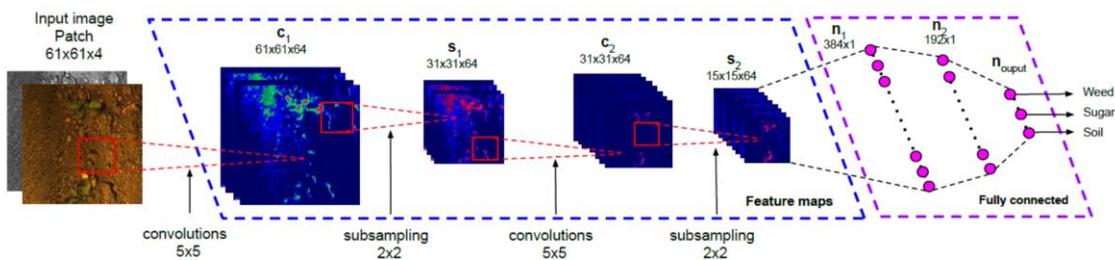


Figura 43 Arquitectura de Red Neuronal Convolutiva *cNET*

Fuente: (Potena, Nardi, & Pretto, 2016)

CAPÍTULO III

DISEÑO MECÁNICO DEL PROTOTIPO

3.1 Diseño del Robot Delta

Para dimensionar los componentes del robot tipo Delta Clásico, es necesario conocer el campo de trabajo que debe cubrir el robot, para ello se parte de la información del cultivo de maíz, donde se tiene que la distancia entre hileras según él (SINAGAP, 2015) es de 800mm. Considerando que la trayectoria del robot Delta puede interferir con el cultivo especialmente cuando tenga un desarrollo vegetativo notorio se limita el área campo de trabajo del robot a 600x600mm con lo que no se afectarán las plantas de maíz.

Según (Balmaceda, 2011), para poder encontrar las dimensiones óptimas de un robot tipo Delta, en función del volumen de trabajo deseado, se puede utilizar un algoritmo basado en el barrido de las variables, dándoles las condiciones propias que debe cumplir un brazo tipo Delta, las cuales son:

$$L1 < L2 \quad (1)$$

$$R \geq r \quad (2)$$

$$p^k \in W_p \quad (3)$$

Dónde:

L1: es la longitud del eslabón conectado a la base fija

L2: es el eslabón conectado en la base móvil

R: es la distancia desde el centro de la parte fija hacia la unión entre el eslabón **L1** y el actuador

r: es la distancia desde el centro de la parte móvil hasta su unión con **L2**

p^k: Son todos los puntos que forman los vértices del volumen de trabajo definido.

W_p: Es el volumen de trabajo.

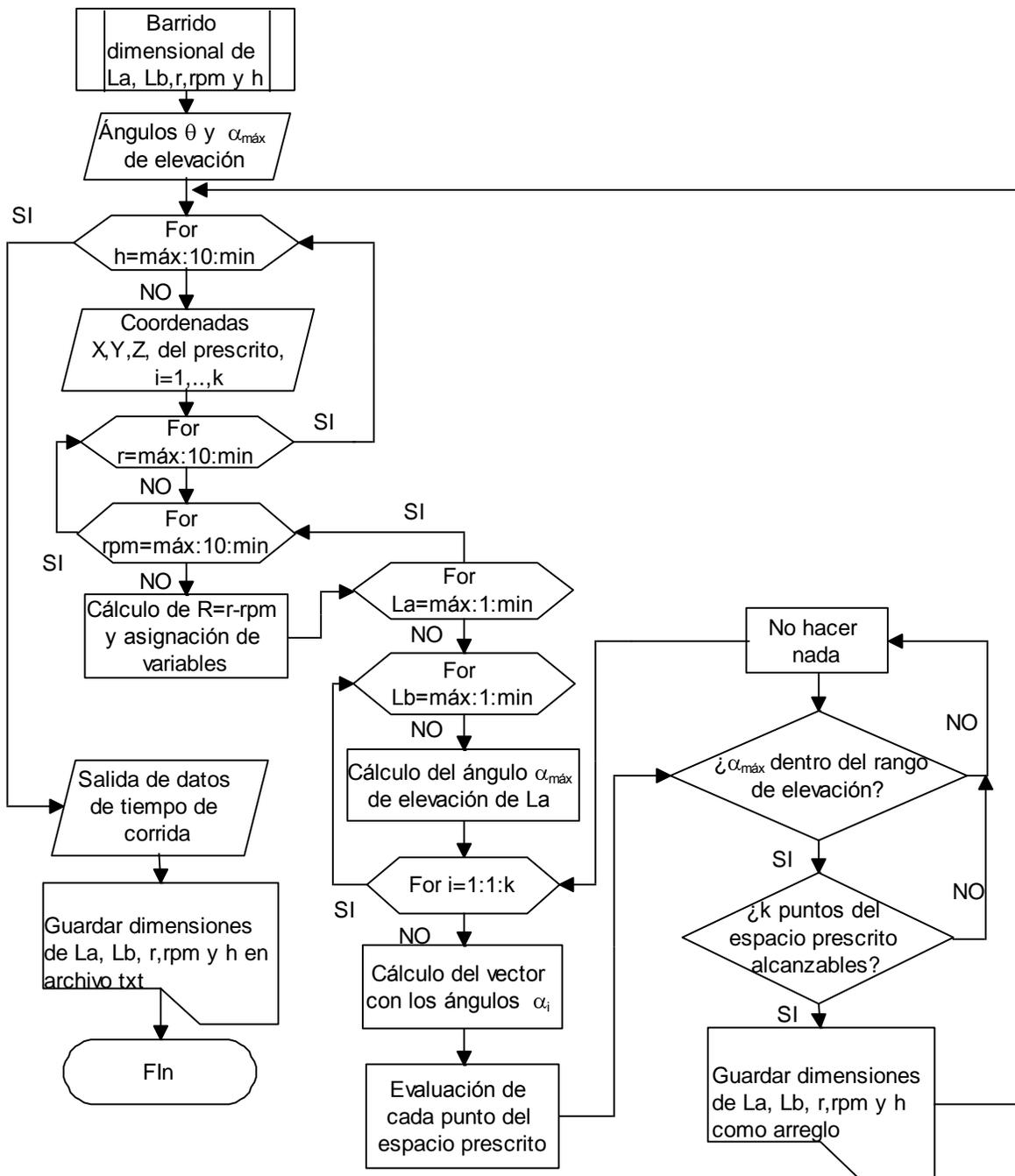


Figura 44 Algoritmo del cálculo de dimensiones óptimas del robot Delta

Fuente: (Balmaceda, 2011)

El algoritmo establece rangos de valores para cada dimensión, y comprueba que cada punto de los vértices del volumen de trabajo satisfaga la cinemática inversa del robot, cuando se cumple esta condición los valores de las dimensiones son guardadas en un vector. Una vez hecho el barrido completo, se

busca el vector que tenga los valores mínimos, de esta forma se obtiene los valores óptimos. Para ejecutar este algoritmo se utilizó Matlab®, el cual entrega como resultado las siguientes dimensiones:

Tabla 3
Dimensiones de las partes del robot Delta

Parte del Robot	Variable	Valor [mm]
Eslabón corto	L1	176
Eslabón largo	L2	500
Radio Base Superior	R	96
Radio Base Inferior	R	80
Altura mínima	Hmin	150

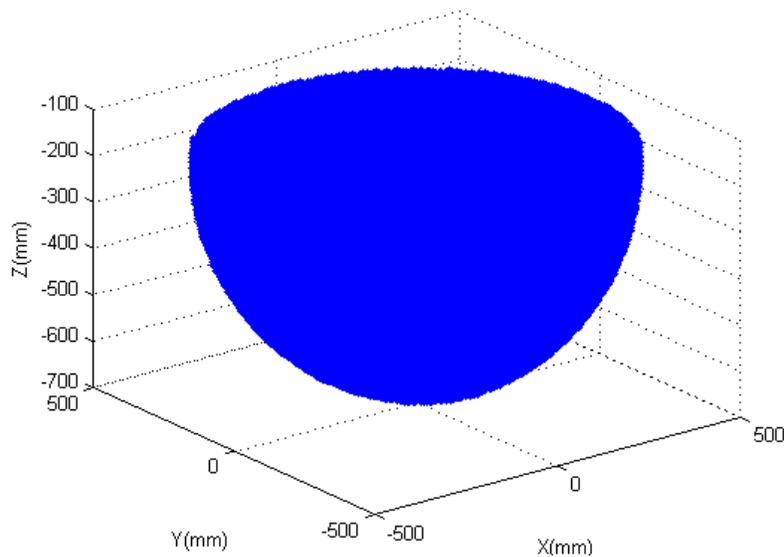


Figura 45 Volumen de trabajo del robot Delta

En la Figura 45 se muestra el volumen de trabajo del robot Delta obtenido en Matlab®, a partir de las dimensiones obtenidas, con dicho volumen se procede a verificar que contenga al campo de trabajo planteado para la deshierba del maíz.

Como se puede observar en la Figura 46, el plano lateral y superior del volumen de trabajo del robot Delta, se aprecia que sí posee las dimensiones necesarias para realizar la acción de deshierba.

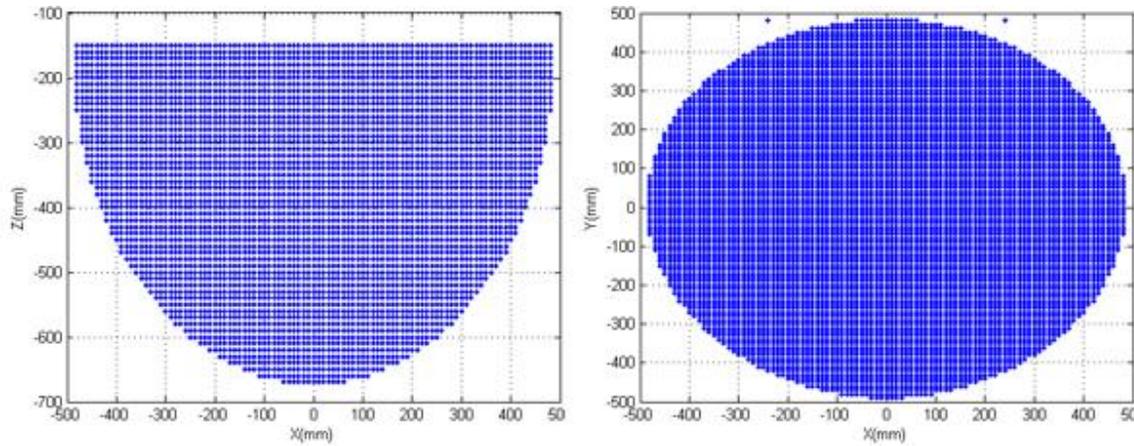


Figura 46 Vista del plano XZ y XY del volumen de trabajo del robot Delta

3.1.1 Selección de los componentes del robot Delta

a. Eslabón corto

Es el de menor longitud y está unido de un extremo con el servomotor y del otro extremo con el eslabón de mayor longitud. Como este elemento va a estar sometido a esfuerzos, el material a usar será perfil de aluminio 6063 de 1.5 mm de espesor. Su diseño se puede observar en la figura 45.

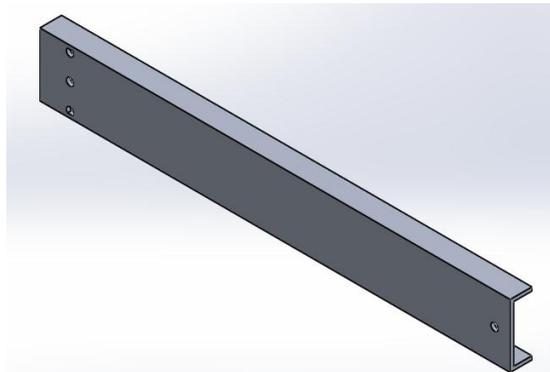


Figura 47 Diseño CAD del eslabón corto.

Para garantizar un diseño seguro es necesario realizar el análisis estructural, para ello se realizarán los cálculos correspondientes y se utilizará como apoyo software de diseño, donde al utilizar para sus cálculos el método de Análisis de Elementos Finitos (FEA), se puede obtener dando como resultado:

- Un análisis de esfuerzos de Von Mises, el cual permite encontrar el esfuerzo máximo al que puede estar sometido un elemento sin romperse, en función de las fuerzas externas aplicadas. Este valor se debe comparar con el esfuerzo de límite elástico correspondiente al material utilizado aplicando la siguiente expresión:

$$\sigma_{vonMises} < \sigma_{LímiteElástico} \quad (4)$$

- Factor de Seguridad (F.S.), que nos permite saber si un diseño es seguro cuando su valor es superior a 1.
- Una referencia aproximada para saber si el factor de seguridad obtenido es el adecuado es a través de la siguiente figura para el diseño de flechas según la norma ANSI – ASME (Díaz del Castillo, 2017).

Tabla 4
F.S. para diseño de flechas según la norma ANSI-ASME

Condiciones de carga	Factor de seguridad F.S.
Cargas estables	2.0
Cargas con impacto moderado	3.0
Cargas con impacto fuerte	4.0

Fuente: (Díaz del Castillo, 2017)

Para analizar el eslabón corto primero se le aplica una sujeción en la parte del acople donde se encontrará el servomotor, luego se le ubica una fuerza externa en el otro extremo correspondiente al peso de los elementos del brazo Delta y del efector final, finalmente se le ubica su propio peso.

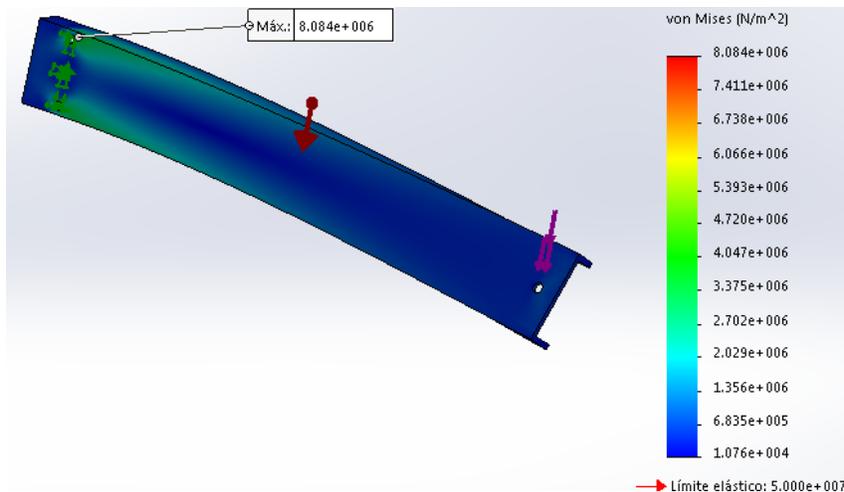


Figura 48 Análisis de Von Mises del eslabón corto del robot Delta

Al aplicar la ecuación 4 se tiene:

$$8.084 [MPa] < 50 [MPa]$$

Una vez comprobado que el análisis de la tensión de Von Mises, cumple satisfactoriamente con la condición, el siguiente paso es comprobar el valor del factor de seguridad (ver figura 49).

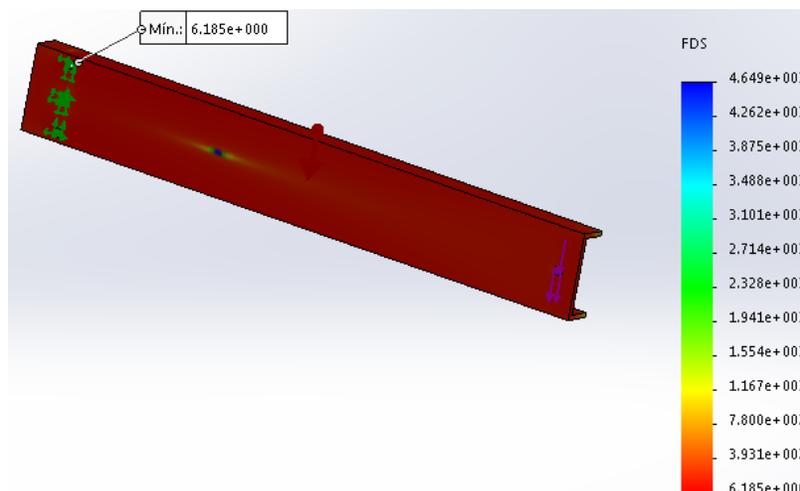


Figura 49 Factor de Seguridad del eslabón pequeño del robot Delta

Como muestra la gráfica, para este elemento se tiene un factor de seguridad de 6, el cual se le considera alto, por lo cual se demuestra que el elemento posee un diseño seguro y que podrá cumplir su función adecuadamente.

b. Eslabón largo

Cada brazo del robot Delta posee dos de estos eslabones, en forma de un paralelogramo, el material para este elemento debe ser resistente y liviano, por lo que se eligió varilla roscada A304 m4, con un paso de 0.7 y para otorgarle rigidez estructural y protección se usó un tubo de Aluminio 6063 de 5/16 como refuerzo para evitar flexiones al momento de utilizar el efector final.

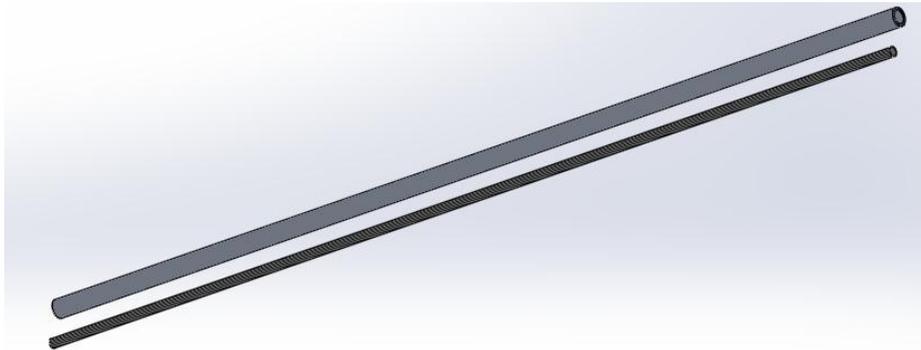


Figura 50 Eslabón Largo (Varilla roscada m4) y Tubo de Aluminio de 5/16

c. Base Superior

La base superior es en donde se sostienen los servomotores y debe ser lo suficientemente fuerte para soportar el peso de todos los elementos del robot Delta. Además, es esta parte a la que se debe acoplar a la estructura de la plataforma. El material seleccionado para este elemento es Aluminio de 2mm de espesor

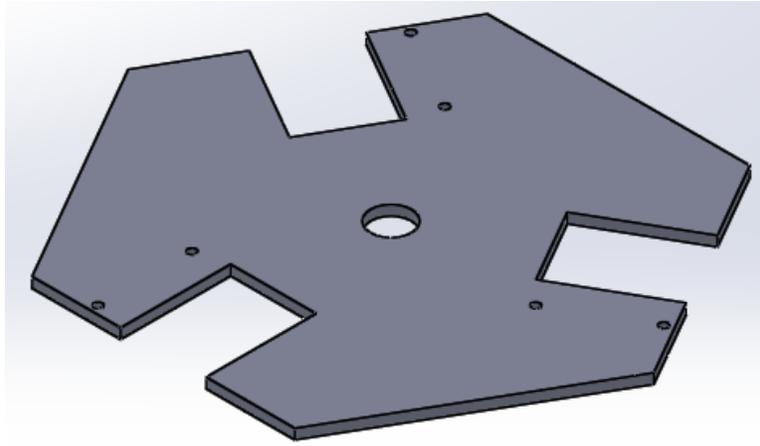


Figura 51 Diseño CAD de la base superior.

Para su análisis se procede a colocar una sujeción de geometría fija en la parte superior de la base superior y se ubica la fuerza correspondiente a todos los elementos del robot tipo Delta, incluido el propio peso del elemento, esta fuerza se le ubica en los taladros donde serán ubicados los servomotores.

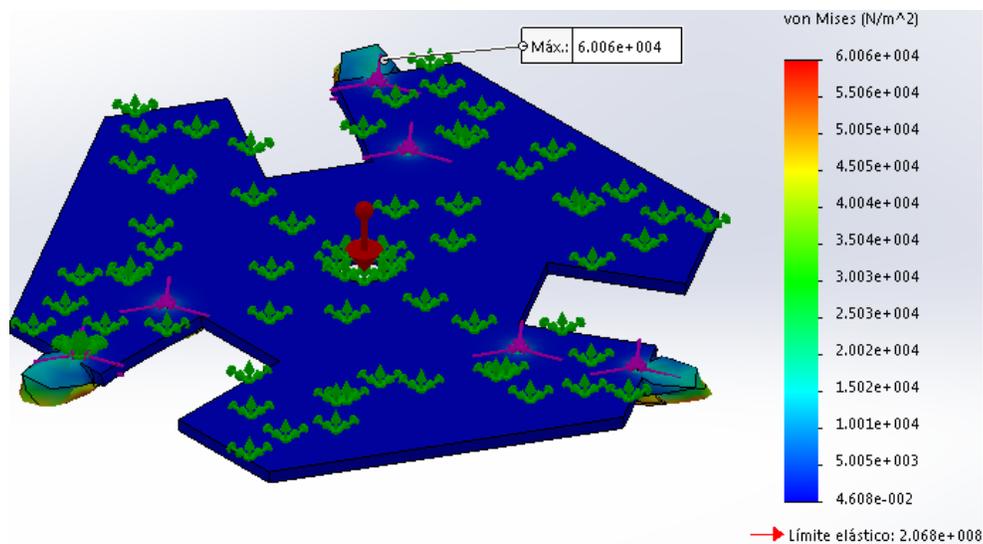


Figura 52 Análisis de Von Mises de la base fija del robot Delta

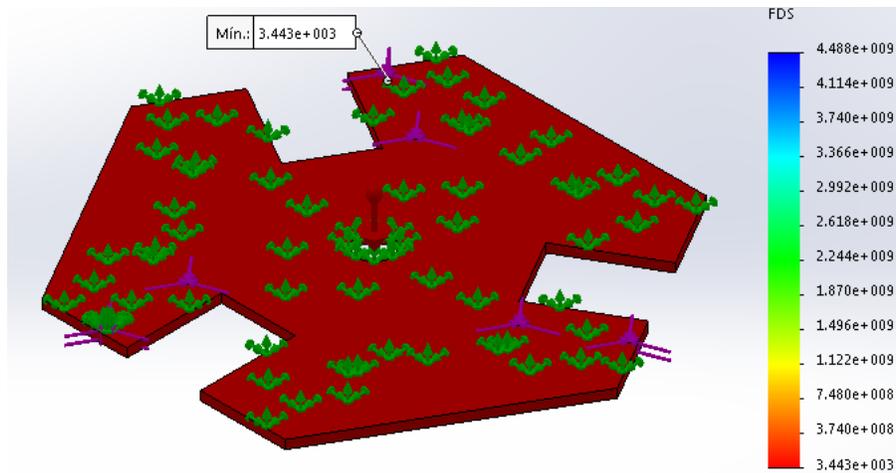


Figura 53 Factor de seguridad de la base fija del robot Delta

Al comprobar en la figura 53, el valor de la tensión de von Mises es menor al valor del límite elástico del aluminio, se corrobora que el diseño cumple la Ecuación 4, y como el factor de seguridad es superior a 3 (por soportar cargas con impacto moderado), se asegura que el elemento tiene un diseño seguro.

d. Base Inferior

Es la parte donde se unen los eslabones grandes en que tienen forma de paralelogramo, además es aquí donde se ubica el efector final, por lo que debe tener las ranuras para poder acoplar de forma segura y precisa el motor con su herramienta. El material para su construcción será PLA por su manufactura mediante impresión 3D, el cual posee un límite elástico de 60 [MPa].

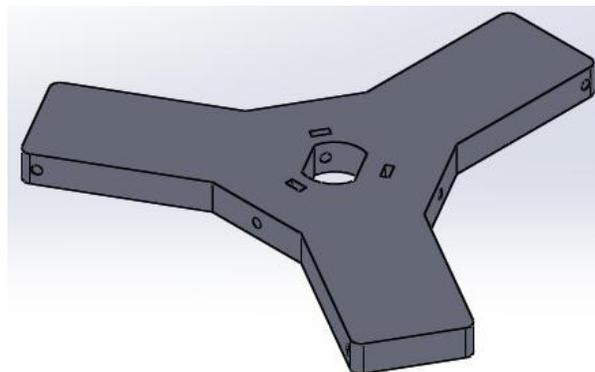
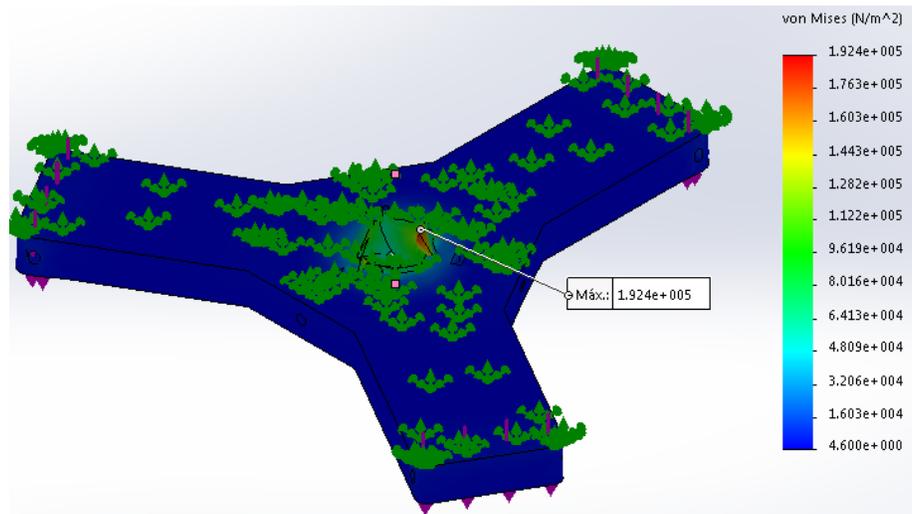
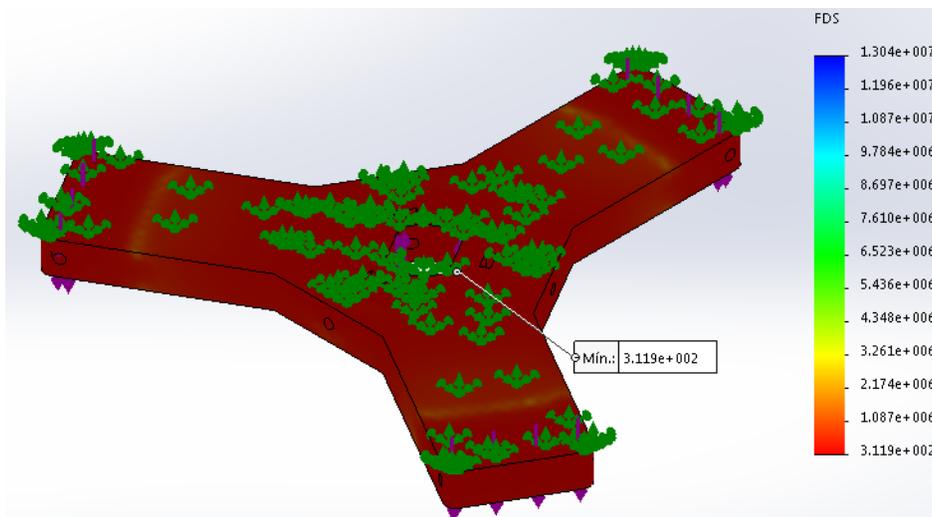


Figura 54 Diseño CAD de la base móvil

Para su análisis estructural se consideró ubicar una sujeción de geometría fija, el peso de los eslabones que se unen con este elemento junto con su propio peso, y una fuerza en sentido opuesto correspondiente a la acción que llevará a cabo la herramienta al realizar la deshierba.



(a) Análisis de Von Mises



(b) Factor de seguridad

Figura 55 Análisis estático de la base fija del robot Delta

Como se puede observar en la figura 55, en los resultados de los análisis estático, los valores de tensión de von Mises, así como el factor de seguridad son adecuados para que el elemento no falle.

e. Rótulas Esféricas

Las rótulas deben tener un rango de movilidad, para que el robot Delta pueda alcanzar todos los puntos de su campo de trabajo sin dificultad, por tanto, el elemento más adecuado resultó las rótulas esféricas Traxxas 5347, puesto que aquí encajan perfectamente la varilla roscada M4 escogida para el eslabón largo.



Figura 56 Junta esférica m4 Traxxas 5347

Fuente: (Banggood, 2017)

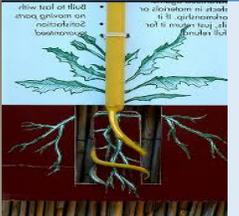
3.1.2 Efecto final

El efecto final es una herramienta giratoria la cual estará encargada de eliminar parcial o totalmente (depende de la edad y tipo de maleza) la mala hierba del cultivo del maíz. Para ello requiere de un motor suficientemente robusto que pueda manejar una herramienta, que ingrese en el terreno para que elimine o por lo menos dañe la raíz de la planta no beneficiosa, con lo que se prevendrá su rebrote.

Ahora, una vez que el robot Delta ha sido definido, la siguiente etapa es buscar un método para eliminar las malezas del terreno en la etapa crítica para el

desarrollo de la planta del maíz que según el INIAP (Yáñez , Zambrano, Caicedo, & Heredia, 2013) es de 0-45 días después de la siembra. Dado que la mala hierba que crece en el terreno después de la siembra de la planta de cultivo es manejable por sus características de tamaño y raíz poco profunda, es posible eliminarla sin que el prototipo requiera utilizar una gran fuerza para la supresión de las mismas a través de un actuador rotativo, por lo que se consideran algunos tipos de herramientas rotatorias convencionales para este fin, estos diseños servirán como base de conocimiento para el desarrollo de una herramienta adecuada que se pueda montar sobre el Robot Delta.

Tabla 5
Herramientas comerciales de deshierba

Nombre de la herramienta	Drill Till ¹	Weed Twister ²	The Awesome Auger ³	Weed Spinner ⁴
Imagen				
Tamaño Original(in)	2.99 x 8.19 x 19.29	13.4 x 4.4 x 4.3	4 x 2 x 42	33 x 4 x 0.5
Peso(lb)	0.9	1.5	2.6	0.8
Forma de arranque	Descenso Vertical	Introducción en raíz	Descenso Vertical	Descenso Vertical

Fuente: ¹ (Drill Till, 2017), ² (Ergonica, 2017), ³ (ASOTV, 2017), ⁴ (Weed Spinner LLC, 2017)

Una vez que se han presentado los modelos de herramientas rotatorias, se pueden considerar los siguientes puntos para el diseño.

- El extremo de la herramienta debe tener punta, disminuyendo la fuerza necesaria del arma para ingresar al suelo.

- La herramienta debe ser lo más simétrica posible, para evitar los efectos de vibraciones al descender al suelo
- La herramienta tiene que contar con un filo de corte, esto es esencial para que al ingresar también se eliminen raíces, como las de mayor diámetro Ej. plantas perennes como el diente de León.
- Considerar en el diseño una parte alargada en la parte superior de la herramienta, para poder eliminar hojas y minimizar los errores de posicionamiento del robot Delta.

Luego de analizar las herramientas más usuales, comprobando su complejidad de construcción y su efectividad al momento de eliminar la maleza, se opta por realizar un diseño de forma parecida a la de Weed Spinner, debido a su fácil manufactura, la cual consiste en una impresión en 3D para la forma y acoples afilados metálicos para la parte de corte y de la punta (ver figura 57).

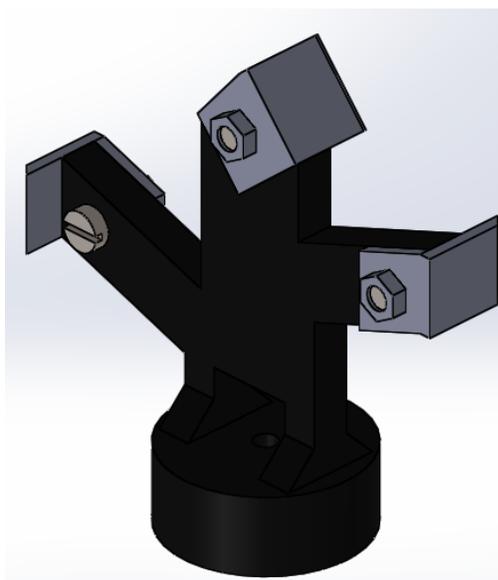


Figura 57 Modelo de la herramienta de corte

a. Cálculos del motor del efector final

Para realizar el cálculo de la potencia necesaria que el motor debe desarrollar para ingresar al suelo de cultivo, se requiere contar con un

procedimiento similar al que se utiliza en las herramientas perforadoras de tierra. Para ello es necesario conocer el valor de resistencia a la compresión del terreno (S_c), para este caso (González, 2004) presenta una forma fácil de para determinar la dureza del terreno (en este caso arcilloso) y su Resistencia a la Compresión, los resultados se muestran en la siguiente tabla.

Tabla 6
Valores y direcciones de identificación de S_c para suelo arcilloso

Consistencia del terreno	Resistencia a la compresión simple en kg/cm^2	Identificación <i>in situ</i>
Muy blanda	$S_c \leq 0,25$	Se puede hundir el puño varios centímetros
Blanda	$0,25 \leq S_c \leq 0,50$	Se puede hundir el pulgar varios centímetros
Media	$0,5 \leq S_c \leq 1$	Se puede lo anterior pero con esfuerzo
Firme(Suelo Rígido)	$1 \leq S_c \leq 2$	Se puede marcar pero con gran esfuerzo
Muy firme (Suelo muy rígido)	$2 \leq S_c \leq 4$	Se raya fácilmente con la uña del pulgar
Dura	$4 \leq S_c \leq 8$	Difícil de rayar con la uña del pulgar
Muy dura	$S_c \geq 8$	Parece roca, pero sin llegar a su S_c

Fuente: (González, 2004)

Como se aprecia en la Tabla 6, los valores pueden estimarse de acuerdo a las reglas *in situ* descritas anteriormente, el peor de los casos que presenta el terreno de pruebas para el prototipo es de tierra dura ($S_c = 8 \text{ kg/cm}^2 = 0,78 \text{ N/mm}^2$), por lo que este será el caso crítico para el normal funcionamiento de la herramienta. Con este valor, se presentan las fórmulas para encontrar la fuerza

de empuje mínima F_{min} suficiente para que se sobrepase el valor de resistencia a la compresión de la tierra y la potencia que debe tener el motor para lograr realizar la penetración en el suelo (Pariante Málaga, 2014):

$$F_{min} = 28.5 * S_c * d_{herramienta} \quad (5)$$

Dónde:

F_{min} : Fuerza de empuje mínima (lb)

S_c : Resistencia a la Compresión (MPa)

$d_{herramienta}$: Diámetro de la herramienta (in)

Utilizando la ecuación (5), se procede con el cálculo reconociendo que el diámetro máximo del arma es de $1^{1/4}$ in, entonces se muestra el siguiente cálculo:

$$F_{min} = 28.5 * (0.78 \text{ MPa}) * (5/4 \text{ in})$$

$$F_{min} = 27.79 \text{ lb} = 0.123 \text{ KN}$$

También es necesario considerar el cálculo de una fuerza de empuje máxima que permita saber con seguridad si el motor cumple para el factor crítico.

$$F_{max} = 2 * F_{min} \quad (6)$$

$$F_{max} = 0.246 \text{ KN}$$

Conociendo este dato, es posible calcular la potencia de rotación (P_r) que permita al actuador realizar su trabajo, para lo cual se presenta la siguiente fórmula.

$$HP_r = 8.55 * 10^{-9} * d_{herramienta}^2 * N_r * F_E^2 \quad (7)$$

HP_r : Potencia de rotación (HP)

$d_{herramienta}$: = Diámetro de la herramienta (mm)

N_r : Velocidad de rotación (rpm)

F_E : Fuerza de empuje (KN)

Para el dato de rotación de la herramienta, se considera 1000 rpm para escoger un motor con una reducción adecuada y la fuerza máxima de empuje, para elegir un motor para el peor de los casos, se calcula la potencia, como se muestra a continuación.

$$HP_r = 8.55 * 10^{-9} * (31.75mm)^2 * 1000 \text{ rpm} * (0.246 \text{ KN})^2$$

$$HP_r = 4,17 * 10^{-4} \text{ hp} = 8.2W = P_r$$

Considerando que el actuador rotativo tenga un 80% de eficiencia (η), se recalcula para obtener el siguiente valor de la potencia necesaria (P_n) real que necesitará el motor, así que se procede de la siguiente forma:

$$\eta = \frac{P_r}{P_n} \quad (8)$$

$$P_n = \frac{P_r}{\eta} = \frac{8.2W}{0.8} = 10.25W$$

Por lo que para la aplicación de la herramienta se requerirá un motor que desarrolle una potencia de 10.25 W. Para lo cual, se eligió un motor Syma X8C, junto con su engrane reductor, el cual presenta las siguientes características que se muestra en la siguiente tabla:

Tabla 7
Descripción de las características técnicas del Motor Syma X8C

Características del Motor Syma X8C	
Tipo de Motor	DC Con Escobillas
Voltaje	7.4V
Velocidad	4400 rpm
Dientes engrane del Eje	11
Reducción	5:1
Dimensiones	20x30x15 mm
Peso	22g

Fuente: (Syma Toys, 2017)



Figura 58 Detalle del Conjunto Motor - Reducción Motores Syma X8C

Como los datos del fabricante no son claros, se procede a realizar una medición de los parámetros eléctricos para comprobar su potencia verdadera, entonces después de 5 pruebas se obtiene el siguiente valor promedio:

Tabla 8

Valores promedios obtenidos de parámetros eléctricos del motor

Características Eléctricas Motor Syma X8C	
Voltaje Promedio	5V
Corriente Promedio	3 ^a

Ahora, se calcula la potencia eléctrica (P_e) del motor, con la siguiente fórmula.

$$P_e = I * V \quad (9)$$

$$P_e = 5V * 3A = 15W$$

Al igual que en el anterior caso, se considera que la eficiencia del motor es de un 80%, por lo que la potencia mecánica final del motor (P_{fm}) se calcula con la ecuación (8):

$$P_{fm} = P_e * \eta \quad (10)$$

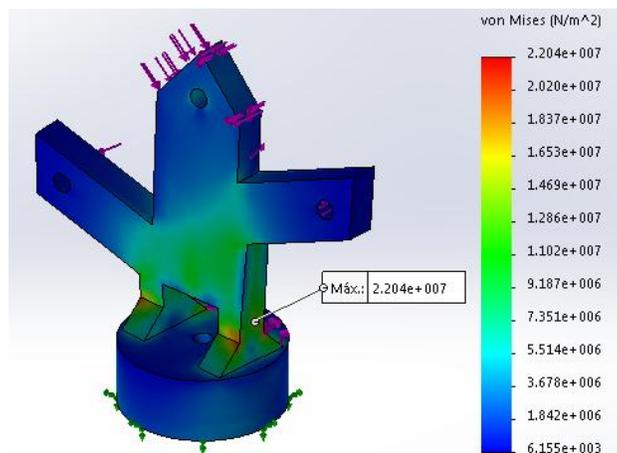
$$P_{fm} = 15W * 0.8 = 12W$$

Conocidos estos datos, entonces se puede asegurar que el motor puede cumplir con la función de ingresar en el suelo de cultivo con la herramienta, como

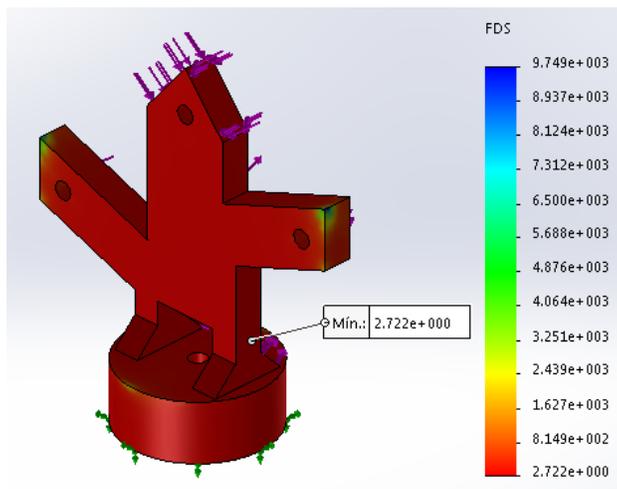
siguiente paso es necesario saber si la impresión 3D con los refuerzos metálicos soportará el trabajo que desarrolla el motor.

b. Análisis estático de la herramienta

Para comprobar que la impresión soporta las fuerzas a las que estará sometida la herramienta, se procede a realizar el respectivo análisis estático del elemento.



(a) Análisis de Von Mises de la herramienta de corte



(b) Facto de Seguridad

Figura 59 Análisis estático la herramienta del robot Delta

Como se puede observar en la figura 59, la tensión de von Mises es inferior al límite elástico del PLA que es de 60MPa, además se tiene un factor de seguridad de 2.7 el que demuestra que el diseño es seguro para esta aplicación.

3.1.3 Selección de los servomotores del Robot Delta

a. Cálculo del Torque Máximo

Para su cálculo, se parte de la definición de torque y se lo analiza en el servomotor con referencia al eslabón al que se encuentra acoplado (ver figura 60).

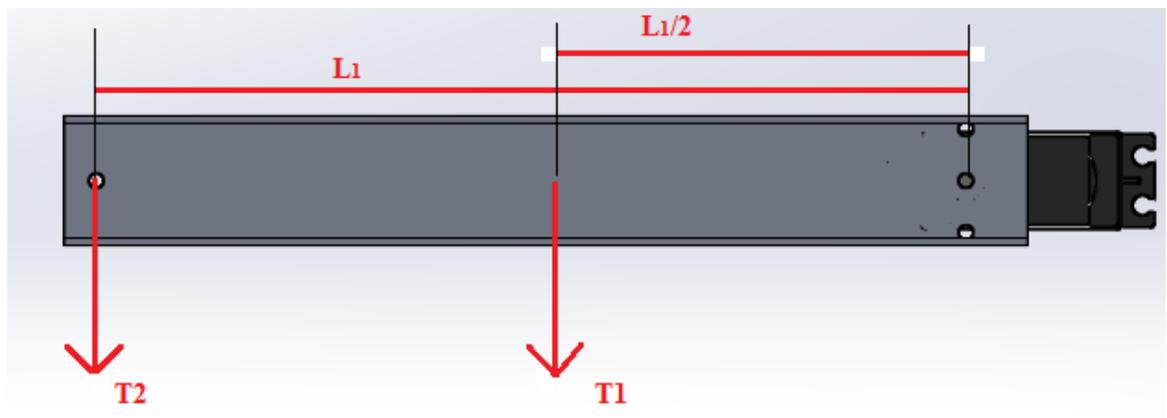


Figura 60 Torques en eslabón 1 y el Servomotor

Se parte de la definición de torque (Ecuación 11):

$$T = F * d \quad (11)$$

$$T_{max} = T_1 + T_2 \quad (12)$$

Para poder estimar un torque máximo, se considera que el extremo del eslabón corto soporta el peso de los dos eslabones largos ($W_{eslabones L.}$) más el peso de toda la base móvil, incluido el efector final ($W_{Base móvil.}$). Además se considera que en el centro del eslabón corto se encuentra su propio peso ($W_{Eslabón c.}$) entonces al reemplazar en (12) se tiene:

$$T_{max} = W_{Eslabón C.} * \frac{L_1}{2} + (W_{eslabones L.} + W_{Base móvil}) * L_1 \quad (13)$$

$$T_{max} = 0.065 \text{ kg} \left(9.81 \frac{\text{m}}{\text{s}^2}\right) * \frac{0.176\text{m}}{2} + (0.1\text{kg} + 0.453 \text{ kg}) \left(9.81 \frac{\text{m}}{\text{s}^2}\right) * (0.176\text{m})$$

$$T_{max} = 0.056 + 0.954 \text{ Nm}$$

$$T_{max} = 1.01 \text{ Nm} = 10.30 \text{ Kg cm}$$

Con el valor teórico calculado del torque, es necesario elegir un servomotor que pueda cumplir estas características, por lo que se presenta una matriz de selección de Servomotores que permitirá elegir el actuador más adecuado.

Tabla 9

Matriz de Selección de los Servomotores para el Robot Delta

Métrica	Importancia	Unidades	Dynamixel AX-12a	Tower Pro MG995	HS-775HB	Tower Pro MG996R
1 Torque	5	Kg-cm	15.2	8.5-10	11-13.2	10.5-13
2 Voltaje	3	V	12	4.8-7.2	4.8-6	4.8-6
3 Corriente máxima	4	A	1.5	0.1	0.28	0.1
4 Rango	5	Grados	300	120	180	180
5 Costo(2017)	4	Dólares	45	16	46	12.5
6 Peso	3	Gramos	56.4	55	110	55
7 Velocidad sin carga	3	Rpm	59	76.9	50	50

Fuente: (TrossenRobotics, 2017)¹ (Prometec, 2017)² (Servodatabase, 2012)³ (Makerlab-electronics, 2017)⁴

Como se puede apreciar en la tabla 9, el servomotor que mejor se ajusta a las necesidades del robot, en cuanto a torque, voltaje, rango de operación, corriente de consumo y costo es el Tower Pro MG996R (ver figura 61).



Figura 61 Servomotor Tower Pro MG996R

Fuente: (Makerlab-electronics, 2017)

b. Disposición final de los componentes

Una vez dimensionados los componentes necesarios, en la figura 62 se presenta el modelo completo del robot tipo Delta, el cual debe ser transportado por la plataforma móvil, para cumplir su cometido de eliminación de malezas.



Figura 62 Ensamble del robot tipo Delta

3.2 Diseño de la estructura de la plataforma

Una vez finalizado el diseño del robot Delta, se debe considerar una plataforma móvil que pueda transportar la estructura de este componente y que

pueda atravesar el terreno de cultivo sin mayores sobresaltos, para ellos es necesario desarrollar un robot móvil. El diseño de la estructura de la plataforma debe cumplir con algunas condiciones de diseño entre las cuales se encuentran:

- Debe ser capaz de transportar al robot tipo Delta a una altura adecuada.
- Las ruedas deben estar en una posición segura para las plantas de maíz.
- Debe tener el espacio suficiente para ubicar todos los componentes necesarios para el funcionamiento del prototipo.
- Debe tener suficiente robustez para atravesar terreno de forma irregular.
- Debe tener un soporte para poder ubicar la cámara en una altura adecuada.
- Debe tener una carcasa de un material resistente y duradero, esta será la encargada de proteger todos los componentes internos del prototipo.

3.2.1 Selección del tipo de configuración del robot móvil

Para seleccionar la configuración del robot móvil se consideran las tres disposiciones que se evaluarán a través de la matriz de selección (ver Tabla 10):

Tabla 10
Matriz de selección de la disposición de la plataforma móvil

	Conceptos						
	Configuración Ackerman			Configuración Triciclo Clásico		Configuración Dirección Diferencial	
Criterio de Selección	Peso (%)	Calificación (0-5)	Evaluación Ponderada	Calificación (0-5)	Evaluación Ponderada	Calificación (0-5)	Evaluación Ponderada
Baja Complejidad	30	3	0.9	4	1.2	5	1.5
Maniobrabilidad		3	0.6	3	0.6	4	0.8
Bajo Costo de Materiales	30	2	0.6	5	1.5	5	1.5
Bajo Peso	20	4	0.8	5	1	5	1
Totales	100		2.9		4.3		4.8

Fuente: (Ollero, 2001)

En la Tabla 10 se consideró los valores que debe cumplir la plataforma robótica, entre ellos el bajo peso y el bajo costo de materiales, por lo que, en la evaluación final, la configuración tipo Direccionamiento Diferencial es la seleccionada para la plataforma principalmente porque posee una buena maniobrabilidad, es de bajo peso y costo. Para esta configuración es necesario la utilización de una rueda loca para la parte posterior y dos ruedas delanteras apropiadas para terreno agrícola con un motor en cada una.

3.2.2 Estructura superior de la plataforma móvil

El diseño de la estructura superior debe tener un adecuado acople para la rueda trasera, una base para las baterías y las ruedas delanteras deben poder ser ajustables a distintos anchos de las hileras de maíz. El material seleccionado fue Acero (NTE INEN 2415), por su resistencia, fácil adquisición, bajo peso y costo.

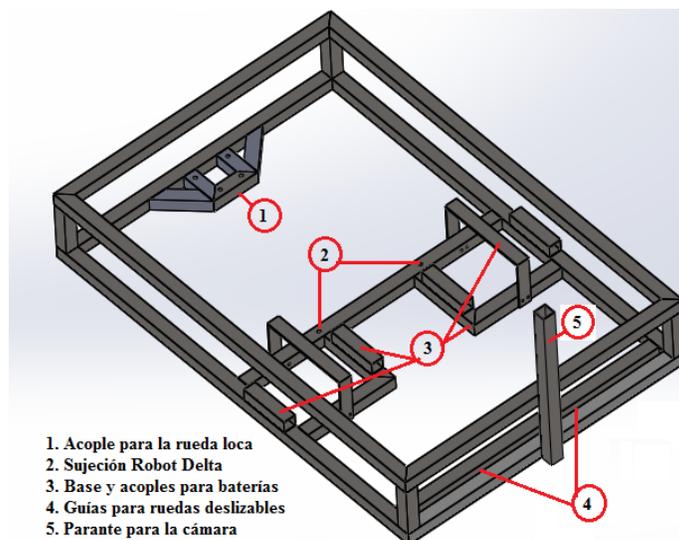
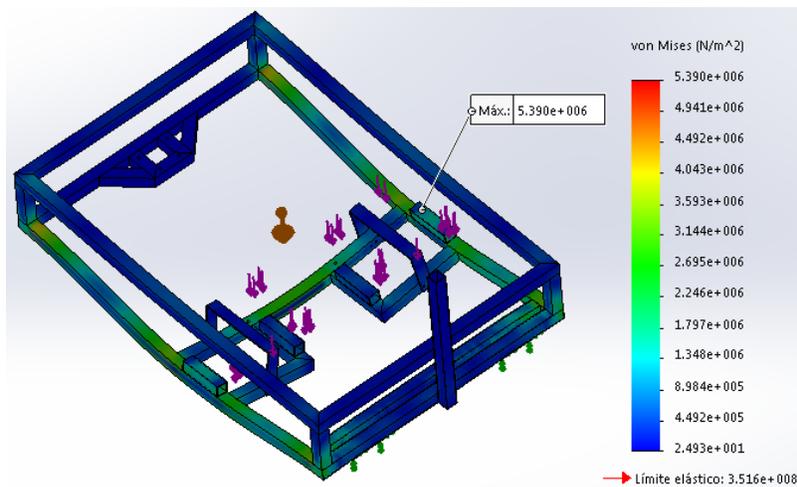


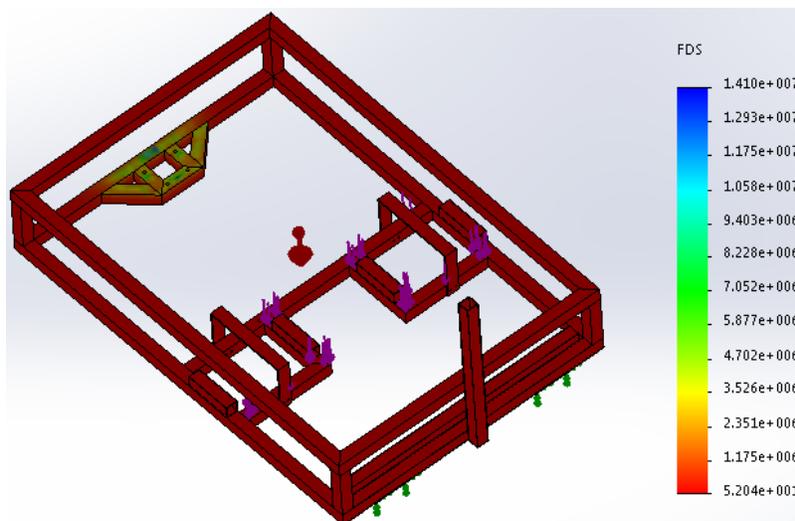
Figura 63 Diseño de la estructura superior del prototipo

Para el análisis estático de este elemento, se consideró una geometría fija en los acoples tanto de la rueda loca como de las ruedas delanteras, luego se aplicó una carga externa equivalente al peso de las baterías y del robot tipo Delta que son las más representativas. Finalmente, se le agregó la gravedad para incluir

su propio peso en el análisis, dando como resultado las gráficas que se muestran a continuación:



(a) Análisis de Von Mises.



(b) Factor de Seguridad

Figura 64 Análisis estático de la estructura superior de la plataforma.

Como se puede apreciar en la figura 64, tanto la ecuación 4 se cumple satisfactoriamente, ($5.39 [MPa] < 351.6 [MPa]$), así como el factor de seguridad tiene un valor de 5.2 que es mayor a 1, por lo que se concluye que el diseño es el adecuado.

a. Soporte de Cámara

Para que la herramienta del prototipo pueda actuar, es necesario primero conocer la posición de las malas hierbas, esto se da a través de un soporte que consta de un tubo cuadrado de 3/4 pulgada doblado en ángulo recto. En su extremo contiene una cámara encargada de enviar las imágenes del terreno de cultivo, para que el robot Delta luego de saber las posiciones pueda actuar cuando se encuentra encima del área procesada, su diseño se muestra en la figura 65.



Figura 65 Detalle del soporte en la estructura del prototipo

b. Carcasa de la estructura superior

La carcasa es necesaria, debido a que hay elementos externos que pueden afectar a los componentes principales del prototipo, además que brinde un fácil acceso al interior del prototipo para su mantenimiento, para ello se ha considerado utilizar tol de acero galvanizado de 0.7mm por su robustez, bajo peso y su propiedad anticorrosiva. En la figura 66, se puede apreciar el cuerpo y la tapa de la carcasa de la estructura superior.

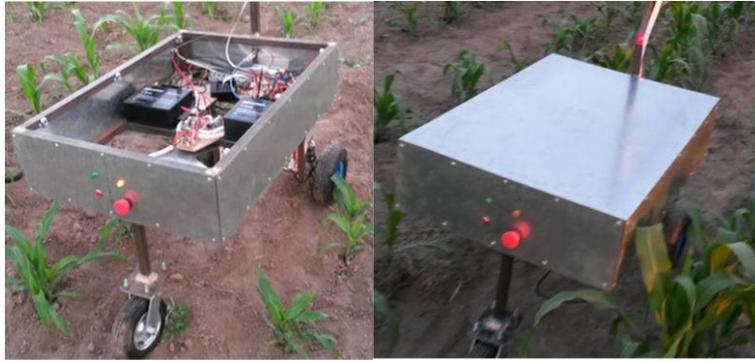


Figura 66 Muestra de la carcasa exterior del prototipo

3.2.3 Ruedas

Las ruedas son parte fundamental de la plataforma móvil, proporcionan tracción al terreno además de conferir cierto nivel de suspensión ante golpes o irregularidades del terreno, para el presente prototipo se han utilizado ruedas estándares de carretilla porque es necesario que sean adecuadas para labores agrícolas, fáciles de conseguir y al ser de cámara de aire no aumentan peso al diseño.



Figura 67 Detalle de la rueda montada con la estructura de suspensión

a. Porta Encoder

Para que la plataforma móvil pueda localizarse en el espacio de trabajo, es necesario de dotarle de sensores para conocer su posición, por lo que se añade a

cada rueda un encoder óptico, para albergar el encoder se diseña un porta encoder que tendrá vista al disco ranurado, su construcción es a través de impresión 3D y su diseño se puede apreciar en la figura 68.

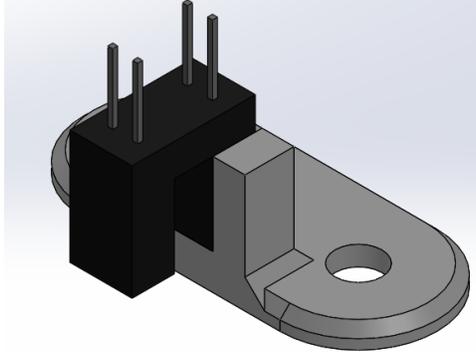


Figura 68 Base para sensor óptico con sujeciones

b. Disco Ranurado

Permite pasar o no luz que genera el sensor óptico. Una parte importante de este disco es su resolución, se optó por un disco con 60 ranuras que corresponde a una mediana resolución, el material usado fue baquelita, su ubicación es en el eje del motor, adjunto a la llanta.

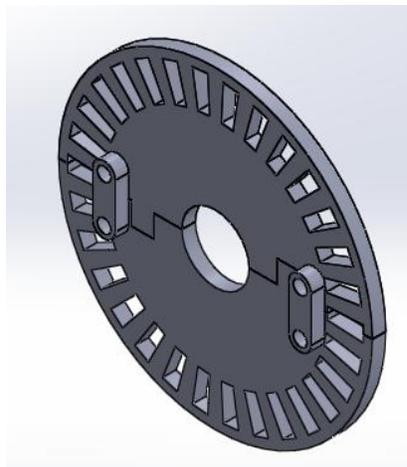


Figura 69 Detalle del disco ranurado

3.2.4 Rueda Loca

Para la selección de la rueda trasera se consideró, que el espesor de la misma debe ser lo suficientemente grande para que la plataforma no se entierre dentro del suelo de cultivo. Esto puede suceder debido al propio peso del prototipo y tiene como resultado la pérdida de tracción y por ende fallos en el cálculo de la localización del prototipo, por lo cual se seleccionó una rueda loca giratoria tipo castor de 8 pulgadas de diámetro y 5 cm de espesor. Para el acople de esta rueda a la estructura, se utilizó tubo cuadrado de acero de 1 pulgada, y platina de acero de 5 mm de espesor, todo el conjunto se puede apreciar en la figura 70.

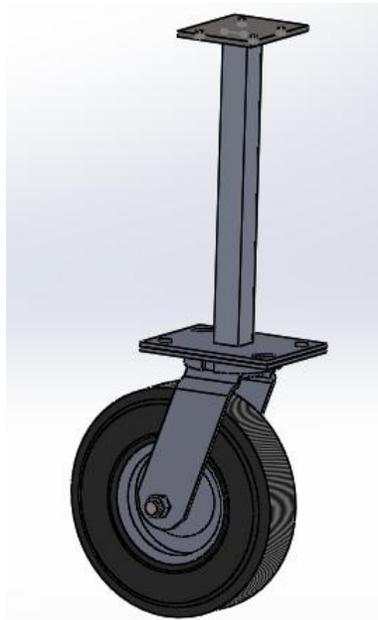


Figura 70 Rueda Loca giratoria y su acople para la estructura superior

3.2.5 Sistema de suspensión

La suspensión en el prototipo es de gran relevancia puesto que su operación se ejecutará en un terreno de cultivo (que normalmente no es completamente plano), por lo que se debe asegurar que exista un adecuado contacto y tracción entre el terreno y las ruedas, además esto evitará oscilaciones o vibraciones que puedan provocar daños futuros a componentes del prototipo.

(Javidan, 2006) menciona una metodología para encontrar la constante de amortiguación del resorte adecuado en función de la frecuencia de suspensión deseada y la forma de colocación en la estructura. Primero se debe tener el peso de la masa suspendida en cada rueda, que viene a ser la diferencia entre el peso total en cada rueda menos la masa ubicada bajo el amortiguador, es decir las ruedas, motores y demás (ver Ecuación 14).

$$M_S = M_T - M_{NS} \quad (14)$$

$$M_S = \left(\frac{24}{3} - 2 \right) kg$$

$$M_S = 6 kg$$

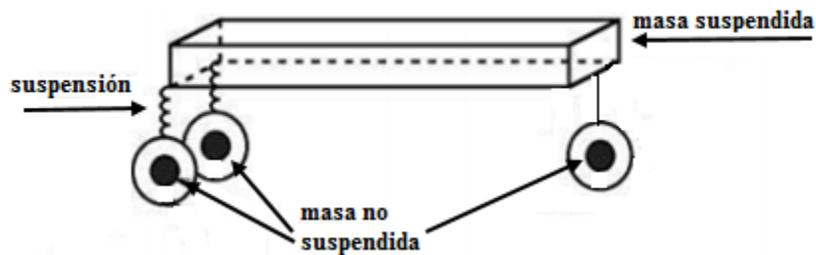


Figura 71 Masa suspendida y masa no suspendida en un sistema móvil.

Fuente: (Garzón, 2012)

En cuanto a los vehículos que deban llevar personas y asegurar su confort se debe procurar que la frecuencia de oscilación sea aproximadamente entre 0.83 y 1.66 Hz, pues son los rangos aceptables para el cuerpo humano (Rocha, 2013), mientras que los de carga pueden llegar hasta 2 Hz, sin afectar la seguridad de la misma, en el caso del prototipo, lo que nos interesa es que tenga un buen agarre (tracción), por lo que para los cálculos se elige una frecuencia intermedia de 1.5 Hz, y se utiliza la expresión de la frecuencia (Ecuación 15) para despejar y calcular el índice de la rueda:

$$f = \frac{1}{2\pi} \sqrt{\frac{WR}{SW}} \quad (15)$$

Donde **f** es la frecuencia, **WR** (Wheel Ratio) es el índice o relación de la rueda y **SW** es valor de la masa suspendida. Luego despejando se obtiene:

$$WR = SW * (2\pi f)^2 \quad (16)$$

$$WR = 6 \text{ kg} (2 * \pi * 1.5\text{Hz})^2$$

$$WR = 1172,51 \text{ kg/mm}$$

a. Índice o relación de movimientos MR (Motion Ratio)

Es una relación que viene dado por las dimensiones entre el centro de giro que va a tener la suspensión, la ubicación del amortiguador y la unión con la rueda, estas medidas se detallan en la Figura 72:

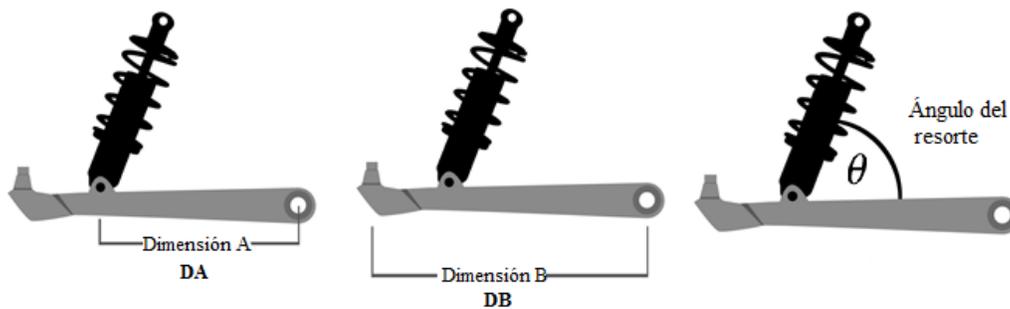


Figura 72 Medidas utilizadas para calcular el índice de movimiento

Fuente: (Ridetech, 2015)

Para encontrar el valor del índice de movimiento se utiliza la Ecuación 17.

$$MR = \frac{DA}{DB} \quad (17)$$

$$MR = \frac{97.5\text{mm}}{119.33 \text{ mm}} = 0.817$$

b. Factor de Corrección del ángulo del amortiguador (ACF)

Debido a que el resorte se ubica con un cierto ángulo de inclinación, es necesario considerar el efecto que tiene esta inclinación, para lo cual se obtiene el factor de corrección del ángulo del amortiguador, el cual es expresado como la función trigonométrica seno del ángulo (ver Ecuación 18).

$$ACF = \text{Sen}(\theta) \quad (18)$$

$$ACF = \text{Sen}(75^\circ)$$

$$ACF = 0.965$$

El factor o constante elástica del resorte, se calcula con la Ecuación 19:

$$SR = K = WR * MR^2 * ACF \quad (19)$$

$$SR = K = 1172.51 * (0.817)^2(1)$$

$$SR = K = 755.24 \frac{lb}{in}$$

El muelle con la constante elástica cercana al valor calculado, es el Ke Zhen 750 lb/in. Con este valor se recalcula el índice de la rueda y obtiene el siguiente valor $WR = 1123.61 \text{ lb/in}$, luego se recalcula la frecuencia y se tiene un valor de $f = 1.46 \text{ Hz}$, la cual se encuentra dentro del rango establecido.



Figura 73 Suspensión Ke Zhen 750

Fuente: (FireCloud Cycles, 2016)

Para la construcción del sistema de suspensión se utilizó tubo cuadrado de acero al carbono de 1 y de 1¼ pulgadas, para de esta manera adaptarlo a las

ruedas delanteras del prototipo de acuerdo a la geometría establecida con el cálculo, quedando el diseño como se observa en la figura 74.

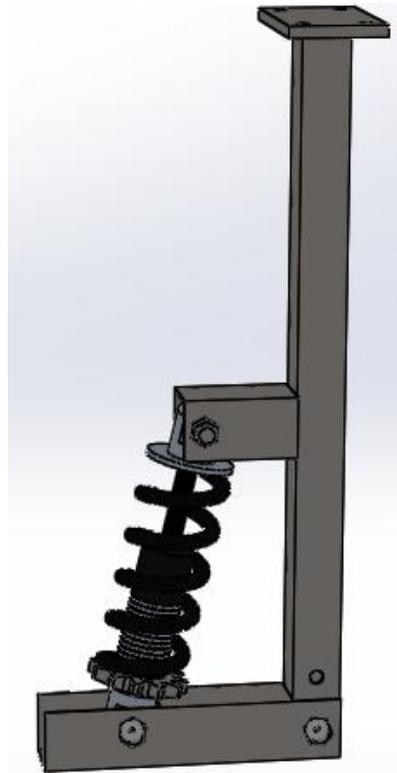


Figura 74 Sistema de suspensión delantera del prototipo

Una vez obtenido el modelo con las respectivas dimensiones del sistema de suspensión se realiza su respectivo análisis estático, para lo cual se hace una sujeción de geometría fija en la parte inferior donde va acoplado el motor. Se ubica un pasador en la unión entre el perfil cuadrado horizontal y el vertical, se agrega la fuerza correspondiente al peso de la estructura que soportará y finalmente se ubica un resorte en el eje vertical con la constante calculada, pero en su componente vertical.

Con este análisis se pueden apreciar en la figura 80 el análisis del esfuerzo de von Mises muestra que este valor es menor al límite elástico, verificando la Ecuación 4.

$$15.9 [MPa] < 351[MPa]$$

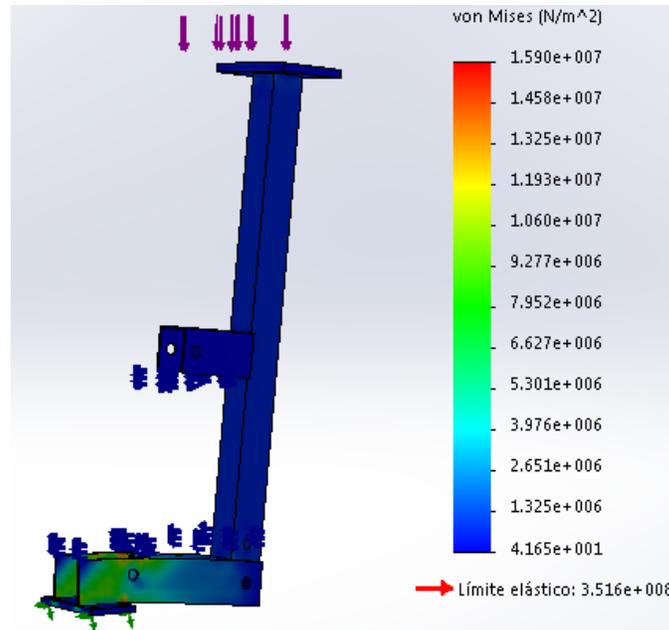


Figura 75 Análisis de Von Mises del Sistema de Suspensión

Así mismo en la figura 81 se puede verificar que el factor de seguridad para este elemento es superior a 1, lo que asegura el diseño y que el elemento podrá realizar su función adecuadamente.

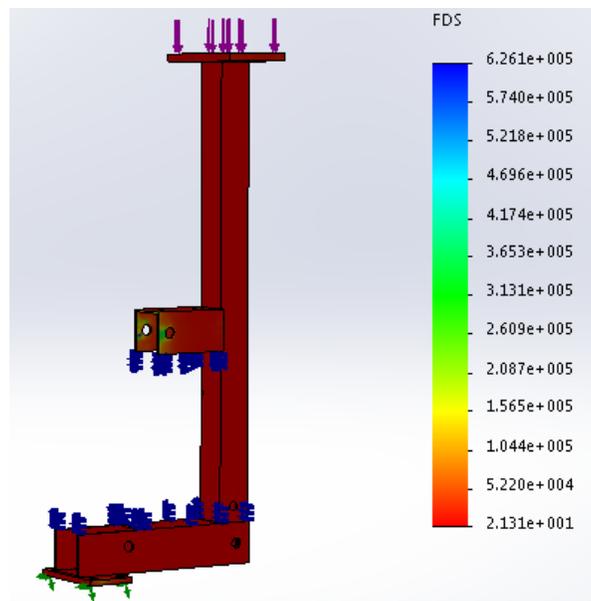


Figura 76 Factor de Seguridad del Sistema de Suspensión.

3.2.6 Selección de los motores

Para realizar la selección de los actuadores eléctricos que permitan el movimiento del prototipo, hay que considerar que el prototipo es móvil, por lo que su alimentación es a base de baterías, entonces son necesarios motores de corriente continua, en la Tabla 9 se realizará la comparación de los accionadores eléctricos normalmente utilizados en el robots móviles según (Torres, Pomares, Gil, Puente, & Aracil, 2002)

Tabla 11
Matriz para Selección del tipo Motores del Prototipo

Selección del tipo Motores							
	Motores DC con Escobillas			Motores DC sin Escobillas		Motores a pasos	
Criterio de Selección	Peso (%)	Calificación(0-5)	Evaluación Ponderada	Calificación (0-5)	Evaluación Ponderada	Calificación (0-5)	Evaluación Ponderada
Alto Torque	35	5	1.75	3	1.05	3	1.05
Facilidad de Control	25	5	1.25	5	1.25	2	0.5
Facilidad Mantenimiento	25	3	0.75	5	1.25	4	1
Disponibilidad	15	5	0.75	3	0.45	3	0.45
	100		4.5		4		3

Fuente: (Torres, Pomares, Gil, Puente, & Aracil, 2002)

En base a estos datos de la Tabla 10, se eligen los motores DC a escobillas debido principalmente a su alto torque y su disponibilidad del mercado, además que su control es sencillo. Con los datos proporcionados por el Diseño Asistido por Computador, se realiza la selección de los motores de acuerdo a las características estáticas del prototipo. El peso máximo admisible para el prototipo

será de : $m_t = 40 \text{ Kg}$. Se realiza un análisis general del prototipo, considerando la rueda donde se aloja el motor, que soportará todo el peso del prototipo. Se realiza el análisis en el arranque del mecanismo donde será el caso más crítico para el prototipo:

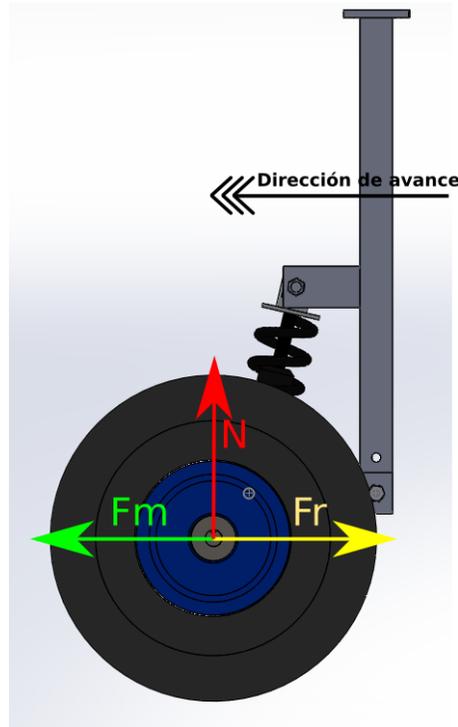


Figura 77 Diagrama de cuerpo libre de la llanta

$$\sum F_x = ma_x \quad (20)$$

En la Ecuación (20), la aceleración del prototipo se considera que desde el reposo debe alcanzar una velocidad que adquiere un Robot Agrícola, el Robot de (Ecorobotix, 2017) alcanza una velocidad de 0.4m/s(promedio), mientras que el Robot de (Naïo Technologies, 2013) alcanza una velocidad de 1.5km/h(0.3611m/s), dado estos datos se elige una velocidad máxima del prototipo de 0.6m/s, que la alcanzará en un tiempo de 1s, por lo que se obtiene la aceleración máxima del prototipo de:

$$Vf_x = Vo_x + a_x * t \quad (21)$$

$$a_x = \frac{Vf_x - Vo_x}{t} = 0.6 \frac{m}{s^2}$$

Ahora se vuelve a la ecuación (20) para obtener el valor final de la fuerza necesaria del motor

$$F_m - F_r = m * a_x \quad (22)$$

Dónde la fuerza requerida del motor es F_m y la fuerza de rozamiento es F_r , ahora, según (HP Wizard, 2017) el factor de rozamiento estático para vehículos fuera de carretera (como tractores) contra el suelo seco es $\mu_s = 0.75$, se considera la normal como el peso total del prototipo, se reemplaza el valor en la ecuación (22):

$$F_m = \mu_s * N + m * a_x \quad (23)$$

$$F_m = 318N$$

Para el dimensionamiento de los motores, se ocupa la fórmula del torque mecánico, se debe considerar el diámetro de la rueda donde estará el motor (r), además de un factor de carga f_c .

$$T_{MT} = f_c * F_m * r \quad (24)$$

Para la ecuación (24) se eligen los valores de factor de carga $f_c = 2$ para un conocimiento de cargas determinable y del entorno de trabajo ordinario (INTI-UE, 2017) y el radio $r = 0,125m$ de acuerdo a la llanta utilizada en el diseño, por lo que se obtiene el torque total requerido para los dos motores:

$$T_{MT} = 79.5Nm$$

Para la selección de los motores, es necesario conocer el valor de la potencia, por lo que se usa la fórmula de la potencia mecánica (25). Como se conoce el dato del radio y la velocidad, se puede expresar la fórmula en los parámetros conocidos a partir de la ecuación de la velocidad angular como se indica en la ecuación (26).

$$P = T_{MT} * \omega \quad (25)$$

$$P = T_{MT} * \frac{V_{max}}{r} \quad (26)$$

$$P = 381.6W = 0.51hp$$

Debido a las pérdidas generadas por la inducción eléctrica y mecánicas, por lo que habrá que considerar en el cálculo una eficiencia del motor, los motores DC promedio tienen como mínimo un 75%-79% de eficiencia, se asume $e = 75\%$, por lo que la potencia corregida obedece a la siguiente relación:

$$P_{corregida} = P * \frac{100}{e} \quad (27)$$

$$P_{corregida} = 508.8W = 0.68 hp$$

Como el valor es considerado general, se divide para dos, con lo que se obtiene la potencia que necesita cada motor para movilizarse dentro de una hilera de cultivo.

$$P_{c/motor} = \frac{P_{corregida}}{2} \quad (28)$$

$$P_{c/motor} = 254.4W = 0.34 hp$$

De acuerdo al valor máximo de potencia que tiene necesita el motor, se procede a realizar la selección de acuerdo a los modelos comerciales existentes.

Tabla 12
Selección de los Motores a la potencia calculada

	Métrica	Importancia	Unidades	Banebots RS775 18V ⁴	Ampflow P40-350 ²	Ampflow M27-150 ³	Ampflow P40-350 ¹
1	Potencia	5	W	246	150	150	150
2	Voltaje	3	V	18	24	24	24
3	Corriente Pico(Rotor Parado)	4	A	130	26.7	26.5	8

CONTINÚA 

4	Eficiencia	5	%	78	70	79	79
5	Velocidad sin Carga	4	Rpm	19500	4900	3800	3800
6	Peso	3	Kg	0.302	4.5	1.36	1.36
7	Reducción Mecánica	3	No	No	No	No	No
7	Precio(Al 2017)	5	\$	17	67	34	34

Fuente: ^{1,2,3} (Ampflow, 2017), ⁴ (Banebots, 2017)

El motor seleccionado fue el Banebots RS775 18V, por presentar las mejores condiciones, sin embargo el torque requerido es mucho mayor al de este motor y las revoluciones por minuto(rpm) son extremadamente altas, por lo tanto es necesario la utilización de una caja reductora para poder aumentar el torque y reducir las rpms. Ventajosamente esta marca de motores también cuenta con cajas reductoras apropiadas para el motor elegido, por lo que la caja que mejor se ajusta a nuestras necesidades es la que posee una reducción de 152.88:1. Estos elementos se muestran en la figura 78.



Figura 78 Motor Banebots RS775 18V y su caja reductora

Fuente: (Banebots, 2017)

3.3 Ensamblaje del Prototipo

Una vez que ha finalizado el diseño mecánico, se emplean algunos procesos para la construcción de los diferentes componentes, la descripción detallada de cada sub-ensamble se presenta a continuación.

3.3.1 Construcción de las ruedas

a. Sub-ensamble motor-caja reductora

Para realizar la unión de este elemento, es necesario considerar primero el ensamblaje de la caja con el Motor Banebots, para ello se necesita acoplar el engrane al eje del motor a través de una prensa manual, como siguiente paso se engrasa los engranes planetarios de la caja y luego a través de los pernos cabeza Allen se encaja toda la estructura como se ve el a Figura 79.



Figura 79 Ensamblaje del conjunto motor-caja reductora

b. Bocín

Dado que la rueda de carretilla no encaja con el eje del motor, es necesario manufacturar un bocín $1 \frac{1}{4}$ in para que se pueda efectuar la transmisión de fuerza del motor a la rueda. Para el bocín se eligió un acero de transmisión AISI 1018, el bocín fue rectificado en un torno paralelo hasta que encaje en el tubo de la rueda y además se realizó un taladro en su centro de $\frac{1}{2}$ in para acoplar el eje del motor.



Figura 80 Construcción del bocín

c. Ensamblaje rueda

Entonces el bocín es asegurado a la rueda a través de unos puntos por soldadura MIG (Metal Inert Gas), el bocín se aseguró al eje de la caja mediante prisioneros M3, la caja posee una distancia de 15mm con el bocín para dar espacio al disco ranurado y con lo que se asegura la transmisión de movimiento.

d. Disco ranurado y porta encoder

Para el disco ranurado se ocupó baquelita como material principal pues es suficientemente fuerte y delgada para ser cortada, para darle la forma se usó el proceso de corte láser. Para el porta encoder se realizó una impresión 3D de acuerdo a las medidas del sensor y asegurándose que se coloque en la parte libre de la caja reductora.



Figura 81 Proceso de corte láser en baquelita

e. Ensamblaje total de la rueda

Una vez descrito todos los componentes, sólo falta juntarlos todos, en la figura 82 se puede observar de mejor manera cómo quedó la disposición de los componentes:

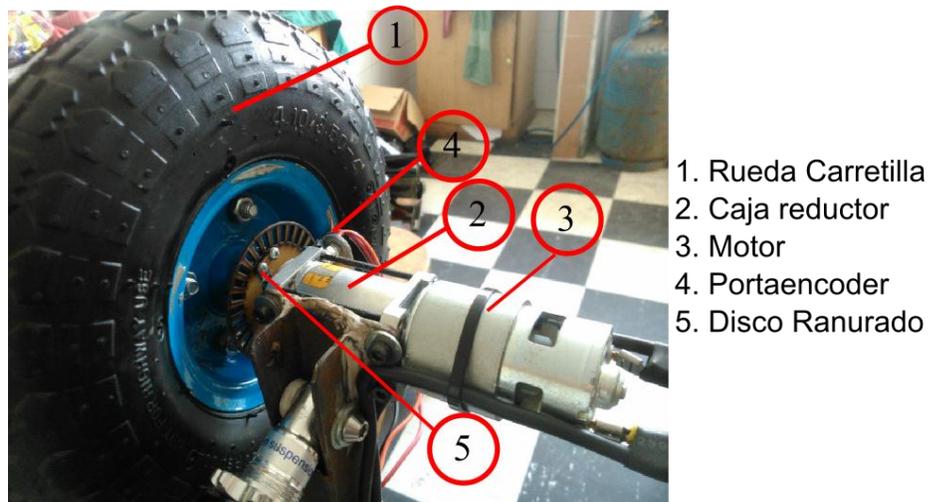


Figura 82 Disposición final de los componentes de la rueda

3.3.2 Construcción del sistema de suspensión y acople a plataforma

a. Construcción acople plataforma – sistema suspensión

Para acoplar la plataforma con las ruedas en primer lugar, es necesario contar con un elemento que conecte a la plataforma y el sistema de locomoción de las ruedas, entonces se usó una platina de 1 ½ in x 5mm de espesor con cuatro taladros, pues como se mencionó en la subsección 3.2.2., que las ruedas serán ajustables.

b. Construcción sistema de suspensión

Con base en el modelo, se corta tubo cuadrado de 1 in x 1mm de espesor, luego se corta el tubo a dichas dimensiones y se realiza los taladros correspondientes, después se lo ubica a los ángulos de diseño. Cuando este cuadrado el diseño, se suelda a través del proceso de Soldadura MIG porque es un espesor muy delgado para otros tipos de proceso. Luego se acopla el amortiguador para finalizar el ensamble de este subsistema. Quedando como se observa en la figura 83.

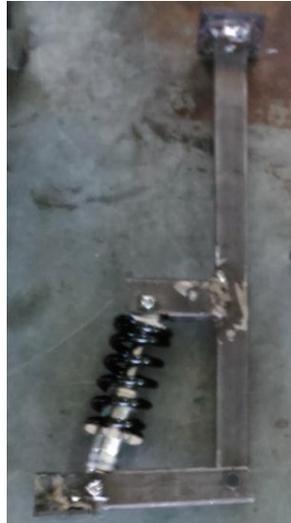


Figura 83 Construcción del sistema de suspensión

c. Construcción de la rueda loca

Para construir la base para la rueda loca, fue necesario usar tubo cuadrado de 1in, además de usar dos platinas de 100x130x5mm para su sujeción a través de pernos a la base de la rueda y a la plataforma móvil. Una vez finalizada la construcción el ensamble quedó como muestra la figura 84.



Figura 84 Detalle de la rueda loca

3.3.3 Construcción de la plataforma móvil

El proceso de construcción de la plataforma móvil constó de tres etapas, la estructura de la plataforma, el soporte de la cámara y el recubrimiento, los pasos que se siguieron se muestran a continuación:

a. Estructura plataforma móvil

Esta estructura se construyó al cortar tubo cuadrado de 1 in y cada pieza de tubo se posicionó acuerdo al diseño. Para la unión de los todos los componentes se escuadró las piezas y se soldaron con el proceso MIG, además se construyeron los acoples de la batería con platina de 1 in.



(a) Disposición de componentes

(b) proceso de Soldadura

Figura 85 Construcción de la Estructura de la plataforma móvil.

b. Soporte de la cámara

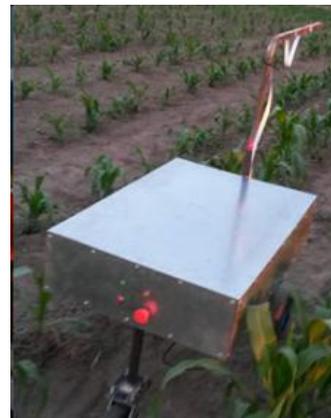
El soporte de la cámara se construye con un tubo cuadrado $\frac{3}{4}$ in para que encaje con lo diseñado en la estructura, se dobla a 90° con una dobladora mecánica y su altura y ancho se debe a las características de la cámara (Cómo se verá en el Capítulo IV), por último, se sujeta a la plataforma móvil a través de dos pasadores para una mayor fijación, obteniendo el ensamble como se muestra en la figura 86.



Figura 86 Detalle del soporte de la cámara

c. Recubrimiento de la plataforma móvil

Cómo último paso de esta sección, se recubre la plataforma móvil para evitar el ingreso de elementos extraños a los componentes de control de prototipo, se usa tol de acero galvanizado para proteger a la estructura de la corrosión propia del campo agrícola. Para cortar el tol se utilizó una cortadora CNC, además se incluye un diseño de tapa del mismo material para poder acceder a los elementos de control fácilmente.



(a) Corte del Tol (b) Recubrimiento final del prototipo

Figura 87 Construcción del Recubrimiento

3.3.4 Construcción del robot Delta con el efector final

a. Base superior

Para la construcción del robot Delta, se parte del diseño de este, la base fue manufacturada a través del proceso de corte por plasma sobre aluminio, entonces la base queda cortada con las medidas exactas.



Figura 88 Proceso de corte por plasma sobre aluminio

b. Acople de los servomotores

Una vez obtenido la base del robot Delta, se acopla los servos a través de ángulos acoplados a la lámina base, los servos son dispuestos a 120° entre sí para cumplir con los parámetros de diseño de la cinemática inversa.

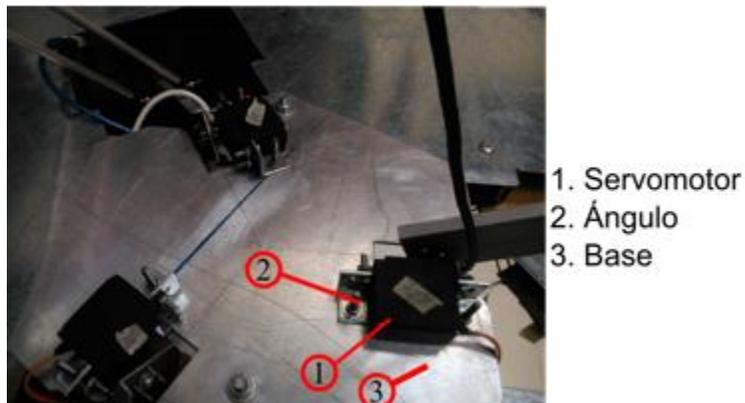
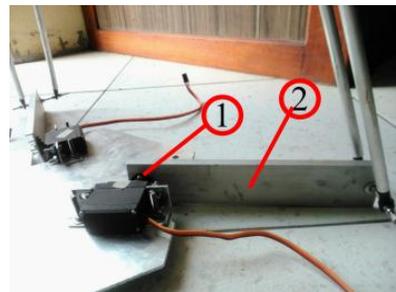


Figura 89 Disposición del acople de los servomotores

c. Acople eslabón corto – servomotor

Para acoplar el eslabón corto al servomotor se utiliza el acople de plástico del que incluye el servomotor, el eslabón corto fue manufacturado al cortar un perfil de aluminio de la medida exacta. El acople del servomotor se ajusta a través de taladros en el eslabón corto.



1. Acople plástico
2. Eslabón Corto

Figura 90 Acople eslabón corto con el servomotor

d. Acople eslabón corto – eslabón largo

Para construir el eslabón largo se utiliza varilla roscada cortada a la medida obtenida del diseño, para darle más rigidez mecánica se añade un tubo de aluminio para que no ceda la varilla ante los golpes, las terminaciones del eslabón largo están compuestas por rótulas esféricas Traxxas que se acopla al eslabón corto mediante una varilla más pequeña y tuercas de seguridad cómo se muestra en la siguiente figura:



1. Eslabón corto
2. Eslabón Corto
3. Varilla Corta
4. Eslabón largo

Figura 91 Detalle ensamblaje eslabón corto y largo

e. Acople eslabón largo – base inferior

Con la terminación en la rótula esférica del eslabón largo, de nuevo se corta una pequeña varilla M3 para que sirva de soporte de la rótula, luego se manufactura en impresión 3D la base y se acopla a los eslabones largos con un agujero que posee para alojar la varilla, la disposición de los elementos se muestra en la siguiente figura:



Figura 92 Detalle subensamblaje Eslabón Largo - Base inferior

f. Efecto final

Ya con la forma del robot Delta, el paso final es acoplar la herramienta, la herramienta consta de los siguientes elementos: Un motor de drone Syma X8C, este se acopla a través de un engrane a una reducción del mismo fabricante y luego a través de un pasador se acopla a la herramienta que fue impresa en 3D, las puntas de la herramienta están reforzadas con pedazos de tol afilados para permitir a la herramienta ingresar al suelo, entonces este ensamblaje queda de la siguiente manera:

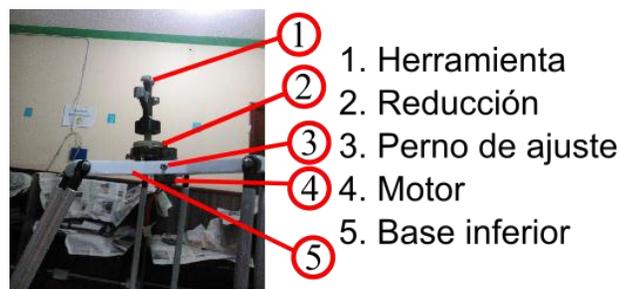


Figura 93 Detalle de la disposición de la herramienta

3.3.5 Ensamblaje total

a. Ensamblaje sistema de suspensión y rueda loca a la plataforma

El sistema de suspensión se acopla a la plataforma móvil a través de la platina con el perno que se ubica en las guías de la plataforma móvil, de acuerdo al ancho de la hilera se ajusta el ancho de las ruedas y se aprietan las tuercas para asegurar. Para acoplar la rueda loca es necesario sólo acoplar la platina con los pernos fijos que se ubican en la plataforma móvil (ver figura 94).



1. Acople plataforma - sistema suspensión
2. Acople plataforma - rueda loca

Figura 94 Detalle ensamblaje con la plataforma móvil

b. Ensamblaje Robot Delta a la plataforma

Para ensamblar el robot Delta a la plataforma se utilizan los dos taladros que se hicieron en la base fija, luego se realizan dos taladros en la parte central de la estructura de la plataforma. Finalmente se utilizó pernos. (Ver figura 95).



Figura 95 Robot Delta ensamblado en la plataforma

c. Ensamblaje soporte de la cámara a la plataforma

Por último, se agrega el soporte que llevará la cámara, se ajusta a la altura adecuada que definen las condiciones del terreno, al igual que su longitud, para el siguiente trabajo, su posición es de acuerdo a lo mostrado en el capítulo IV, la unión es como muestra en la figura 96:



Figura 96 Soporte ensamblado en la estructura

d. Pintura

Una vez unidos todos los elementos, se pinta el prototipo de un color de maquinaria agrícola (el normales amarillo con negro), la pintura permite alertar al usuario de la ubicación del robot y protegerlo de la corrosión así como evitar la concentración de calor en los componentes internos.

Se optó por pintar de color amarillo toda la estructura superior, es decir la carcasa y el parante de la cámara, mientras que se pintó de negro el acople de la rueda loca, así como el sistema de suspensión. Es así que el ensamblaje final del prototipo queda como se puede apreciar en la siguiente figura 97:



Figura 97 Ensamble final del prototipo

CAPÍTULO IV

DESARROLLO DEL CONTROL DEL PROTOTIPO

4.1 Selección de componentes para el control

Para realizar una correcta selección de los componentes primero se debe tener muy en claro el funcionamiento del prototipo. Para lo cual se establecen las funciones principales que deberá cumplir el mismo.

- Capacidad de adquirir las imágenes y procesarlas
- Reconocer y obtener la ubicación tanto de la mala hierba como del maíz.
- Tener un sistema de navegación en base a sensores, para poder seguir trayectorias, acorde a las hileras de los cultivos.
- Controlar la velocidad, sentido de giro de los motores que desplazará la plataforma móvil, con la potencia necesaria.
- Tener energía suficiente para cumplir todas las funciones.
- Controlar los servomotores, encargados de proporcionar el movimiento del robot tipo Delta.
- Tener un panel de control para iniciar el funcionamiento, así como un paro de emergencia e indicadores.

Una vez definidas las funciones del prototipo, se debe seleccionar los elementos necesarios de acuerdo a los requerimientos técnicos y que permitan cumplir satisfactoriamente cada tarea descrita. Dichos elementos se enlistan en la siguiente tabla:

Tabla 13
Elementos para realizar el control del prototipo.

Cantidad	Elemento	Función
1	Microcomputador y su fuente	Procesar imágenes y clasificarlas

CONTINÚA 

1	Cámara	Adquirir imágenes
2	Drivers	Controlar los motores
2	Encoders	Monitorear la posición de las ruedas
2	Baterías	Proporcionar energía al sistema
1	Microcontrolador	Controlar periféricos del prototipo
1	Paro de emergencia	Detener el sistema
1	Botón de inicio	Iniciar las funciones del sistema
2	Luces piloto	Indicadores del funcionamiento

4.1.1 Selección del microcomputador para visión

Este elemento es la de la parte principal para la operación del prototipo, tiene por función procesar, clasificar y enviar datos a la plataforma móvil. Al tratarse de un robot móvil, una característica de la tarjeta es que debe ser transportable. Para la selección de un sistema embebido de visión es importante analizar las características que poseen las soluciones más actuales, como se puede observar en la siguiente tabla:

Tabla 14
Características de tarjetas Controladoras

Característica	Nvidia Jetson ¹	Raspberry Pi 3 ²	BeagleBone Black ³
Procesador	Tegra K1 SOC	ARM Cortex A53	AM335x
Velocidad	2.3Ghz	1.2GHz	1 GHz
RAM	2Gb	1GB	512 MB
USB	2	4	1
Audio	HDMI, Analógico	HDMI, Analógico	HDMI
Video	HDMI	HDMI, Analógico	Mini-HDMI

CONTINÚA 

Compatibilidad con Cámaras Comerciales	Si	Si	Solo cámaras compatibles con formatos comprimidos
Ethernet	10/100/1000	10/100	10/100
WiFi	n/a	WiFi 802.11n integrado	n/a
I/O	7 pines digitales de Entrada/Salida	40 GPIO pins	69 GPIO, LCD, GPMC, MMC1, MMC2, 7 AIN, 4 temporizadores, 4 puertos seriales, CAN0
Tamaño	5" x 5"	3.37" x 2.125"	3.4" x 2.1"
Sistema Operativo	Linux4Tegra OS	Linux	Android, Linux, Windows, Cloud9, CE, etc
Entorno de programación	NVIDIA Nsight, IDLE, OpenEmbedded, QEMU, Scratchbox, Eclipse	Linux, IDLE, OpenEmbedded, QEMU, Scratchbox, Eclipse, QT creator	Python, Scratch, Linux, Eclipse, Android ADK
Costo	192 USD	35 USD	45 SD

Fuente: ¹ (NVIDIA, 2017), ² (Raspberry Pi Foundation, 2016), ³ (Beagleboard.org, 2017)

De las tarjetas analizadas, la más adecuada para el prototipo resulta ser el Raspberry Pi 3, por su costo, su memoria RAM y por alta velocidad de procesamiento, además esta tarjeta permite una interfaz adecuada y compatibilidad con cámaras USB y CSI, aspecto que no cuenta la tarjeta Beagle Bone Black. El aspecto de la tarjeta Raspberry Pi 3 se puede observar en la figura 98.

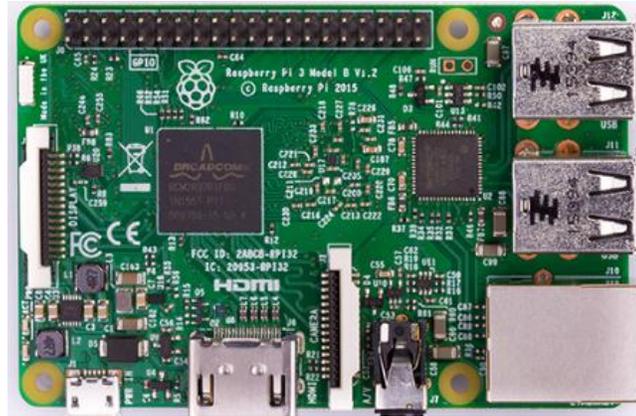


Figura 98 Microcomputador Raspberry Pi 3
Fuente: (Raspberry Pi Foundation, 2016)

4.1.2 Selección de la cámara

a. Selección del sensor

Para la selección de la cámara que será la encargada de obtener las imágenes para el procesamiento, hay que considerar algunos aspectos importantes, primero su compatibilidad con el Raspberry Pi que fue el microcomputador elegido. Otro aspecto importante es la resolución del sensor, aprovechando el conector MIPI CSI que posee el dispositivo (según las especificaciones de (MIPI Alliance, 2014) puede manejar resoluciones FullHD y recomendado para visión artificial), por lo que se elige el módulo Raspberry Pi Camera V2 ya que posee una resolución de 8MP y un sensor Sony IMX219 (Raspberry Pi Foundation, 2017) de gran rendimiento para videos y fotos. En la tabla 15 se puede apreciar las características técnicas del dispositivo.



Figura 99 Detalle del Módulo de Cámara V2 para Raspberry Pi

Fuente: (Raspberry Pi Foundation, 2017)

Tabla 15
Características Técnicas del Módulo de Cámara V2

Característica	Detalle
Sensor	Sony IMX219
Tipo de Sensor	CMOS
Resolución	8MP con enfoque fijo(ajustable manualmente)
Interfaz de Bus	CSI-2
Máxima Resolución	3280 x 2464
Máximos Cuadros por Segundo (A la máxima resolución)	30fps
Dimensiones	23.86 x 25 x 9mm
Máxima temperatura de operación	+60°C
Mínima temperatura de operación	-20°C
Área del sensor de imagen	3.68 x 2.76 mm (4.6 mm diagonal)
Relación del sensor	4:3
Tamaño del píxel	1.12 μm x 1.12 μm
Tamaño óptica	1/4"
Distancia focal	3.04mm
Modos Adicionales	Slow Motion
Correcciones imagen	Control automático de exposición(AEC), Balance automático de blancos(AWB), Calibración automática de nivel negro(ABL), Cancelación de píxeles defectuosos, etc.

Fuente: (Raspberry Pi Camera V2 Camera Module, 2017)

b. Selección de la altura de la cámara

Una vez seleccionado el sensor de la cámara, es necesario ubicarlo a una altura óptima para que el sensor pueda obtener una imagen donde se aprecien las dos hileras del cultivo de maíz.

Entonces cómo se definió en el diseño mecánico, la hilera tiene un ancho de 800mm, por seguridad se dobla ese valor, para que la cámara pueda ver al menos 1600mm, con lo que se abarcaría una dos hileras más las otras dos mitades del ancho de la hilera para luego así poder compensar la trayectoria del prototipo en el control. Lo dicho anteriormente, se puede entender de mejor forma en la imagen 100.

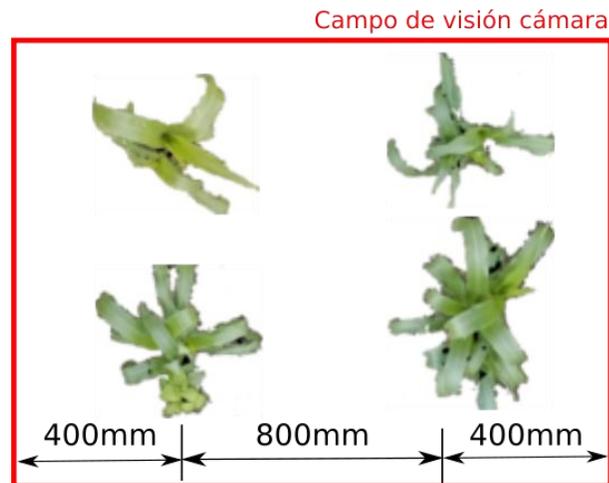


Figura 100 Campo de visión estimado para la cámara

Para poder establecer la distancia requerida, es necesario conocer los datos del conjunto óptico del sensor de la cámara, el sensor tiene una relación de 4:3 donde el área del sensor de imagen es de 3.68x2.76mm y una distancia focal de 3.04 mm (que representa la distancia del sensor al lente), para proceder con el cálculo se debe tomar la siguiente consideración como se muestra en la imagen 101.

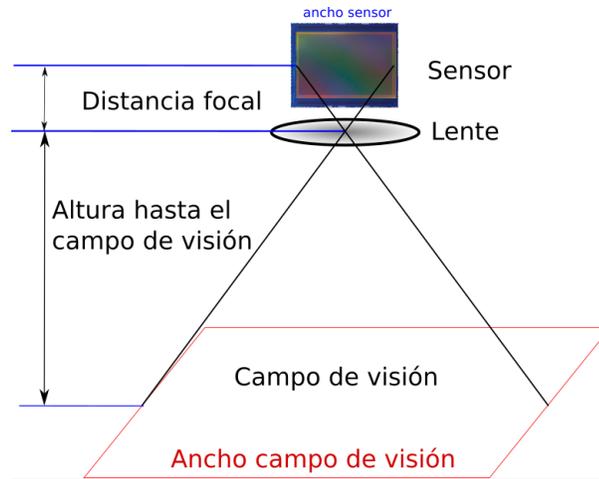


Figura 101 Relación de la longitud focal con la altura de la cámara

Cómo se aprecia en la figura 101, hay una relación entre los triángulos (pues sus ángulos son opuestos por el vértice) de la distancia focal y la altura hasta el campo de visión, por lo que se puede plantear la siguiente ecuación para obtener el ancho de campo de visión:

$$\frac{\text{distanciafocal}}{\text{ancho sensor}} = \frac{\text{altura hasta el campo de visión}}{\text{ancho campo visión}} \quad (29)$$

$$\text{ancho campo visión} = \frac{\text{ancho sensor} * \text{altura hasta el campo de visión}}{\text{distanciafocal}} \quad (30)$$

Entonces, arbitrariamente la cámara se ubica en el soporte a 1260mm de distancia con respecto al suelo, para luego encontrar el ancho del campo de visión con los datos del más grande del sensor(ancho), entonces el ancho de campo de visión (coordenada x) con los datos dados es: 1525mm. Este valor es cercano a la distancia deseada, por lo que se acepta, entonces, se recalcula el valor para el lado más pequeño del sensor con lo que se obtiene el largo del campo de visión (coordenada y). El valor calculado fue de 1143mm en el campo de visión, por lo que se debe asegurar que la cámara se encuentre a la mitad de la distancia con respecto al punto extremo de las llantas del prototipo, para que las llantas no

aparezcan en la imagen, por lo que la posición de la cámara y el campo de visión, quedaría como se muestran en la figura 102 y 103.

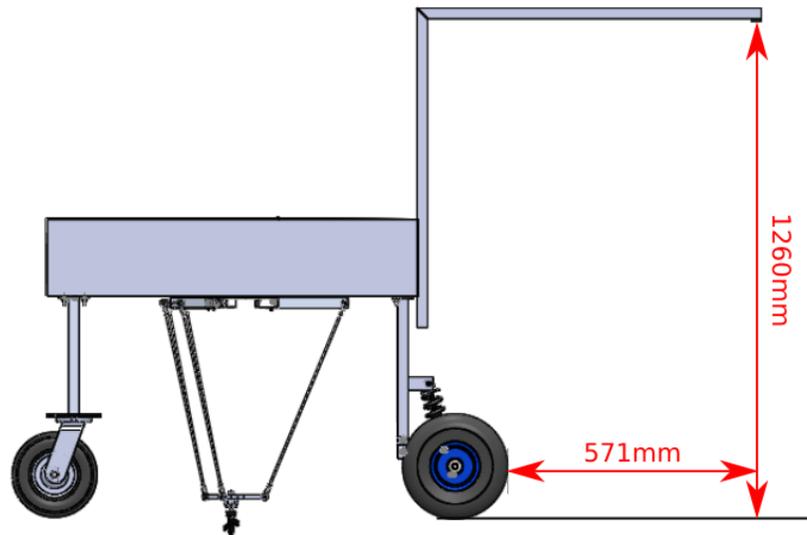


Figura 102 Medidas de ubicación de la cámara en el soporte cámara

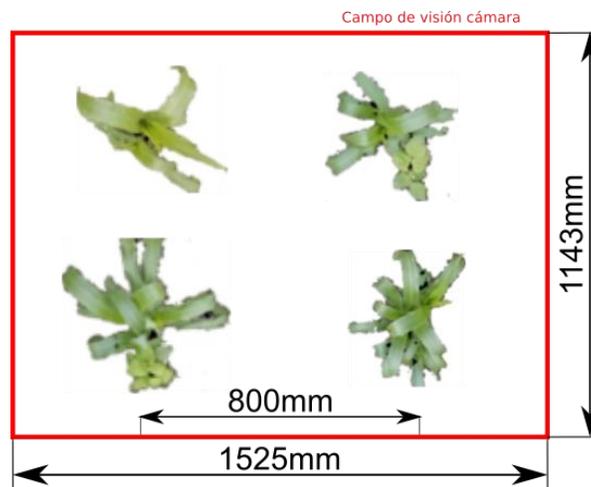


Figura 103 Campo de visión final de la cámara

4.1.3 Selección del microcontrolador

Una vez que ha sido elegido el dispositivo encargado de adquirir y procesar las imágenes, se debe otorgar todos sus recursos para ese objetivo, por lo que es

necesario considerar un sistema integrado que permita controlar los periféricos conectados al prototipo, para los cuales se presenta la tabla 16 donde se relaciona los microcontroladores más populares:

Tabla 16
Características de microcontroladores populares

Característica	Atmega 328 (Arduino Uno)	PIC18F45K50 (Pingüino)	PIC 16F877A
Velocidad	16 Mhz	48 Mhz	20 Mhz
Voltaje	7-12 V	2- 5.5 V	2- 5.5 V
Entradas/Salidas Digitales	14	13	33
Entradas Analógicas	6	5	8
Salidas PWM	6	2	2
Memoria Flash	32k	32k	8k
Memoria EEPROM	1024 bytes	512 bytes	256 bytes
Corriente continua por PIN	40mA	25 Ma	25mA
Protecciones cortocircuito	Si	No	No
Costo	23 USD	9 USD	8 USD

Fuente: ¹ (Arduino, 2017), ² (Microchip, 2001), ³ (TecBolivia, 2013)

Conocidas las características de los tres microcontroladores se elige al Atmega 328 correspondiente a la tarjeta Arduino Uno, debido a que posee un mayor número de salidas con PWM (Modulación de ancho de pulso), necesarias para controlar las velocidades de los motores y posiciones de servomotores, además la tarjeta posee la mayor capacidad de memoria EEPROM, lo cual es

importante para almacenar la información de las coordenadas de las malas hierbas a eliminar.

4.1.4 Selección del controlador de los motores principales

Para la selección del controlador del motor, primero se debe establecer qué tipo de motor DC es, el Motor Banebots RS775 es un motor DC con escobillas, por lo que el controlador del motor debe soportar este tipo de cargas. Como primer paso para elegir el controlador adecuado es verificar si el controlador soporta el voltaje del motor, luego de esto es necesario especificar la corriente continua de operación del motor, como muchos motores de altas potencias no especifican este dato pero según (Coleman, 2017) se puede establecer que la corriente continua de operación del motor es el 25% de la corriente a rotor parado, Coleman también recomienda que el controlador pueda soportar al menos el doble de la corriente de operación, para labores de torque como la subida de pendientes. Por lo que, dado los datos de la corriente a rotor parado del motor elegido es:

$$I_{max} = 16A$$

$$I_{maxcontrolador} = 2 * I_{max} \quad (31)$$

$$I_{maxcontrolador} = 32A$$

Una vez obtenido la corriente necesaria del controlador, se procede a seleccionar el driver adecuado para el motor entre las opciones presentadas a continuación:

Tabla 17
Características de tarjetas controladoras de motores

	Unidades	CYTRON 30A DC ¹	Sabertooth Dual 2x32A Regenerative Motor Driver ²	RoboClaw 2x30A ³
Control Bidireccional	Si/No	Si	Si	Si
Corriente Pico Máxima	A	80	64 (Por Canal)	30(Por Canal)

CONTINÚA 

Corriente Continua	A	30	32(Continua por Canal)	60(Por Canal)
Protecciones	-	Polaridad Reversa	Sobrecorriente y termal	Protección cortes de energía
Frecuencia PWM	Khz	20	30	-
Voltaje del Motor	V	5-30	6-30	6-34
Número de Motores Disponibles	#	1	2	2
Modos de Entrada	-	Entrada 3.3 – 5 V	Análogo, R/C, Serial, USB	USB
Precio	Dólares	33	125	125

Fuente: ¹ (CYTRON, 2017), ^{2,3}, (RobotShop, 2017)

Para cumplir con los requerimientos del motor, se selecciona dos drivers CYTRON 30A DC debido a su bajo precio y a sus características de soportar una alta corriente para la operación en alto torque.

4.1.5 Selección del encoder para los motores principales

Para poder desarrollar la navegación del prototipo es necesaria la utilización de encoders los cuales permitirán conocer la posición del prototipo en movimiento y realizar un control de trayectoria. Por las características físicas del motor, no es posible la utilización de estos sensores ubicados dentro del motor, por lo que se optó por encoders externos, para lo cual se comparan tres tipos de sensores ópticos. Estos sensores deben ser lo suficientemente robustos para que no sean afectados por las condiciones ambientales del lugar de trabajo del prototipo.

Tabla 18
Características de sensores infrarrojos

Característica	QRD1114	TCRT5000L	H20104
Tipo de detector	Optoreflexivo	Optoreflexivo	Fotobarrera
Voltaje (V)	5	5	4.5 -30
Alcance (mm)	0.2-10	0.2 – 15mm	-

CONTINÚA 

Potencia(mW)	100	100	100
Dimensiones(mm)	6.1x4.39x4.65	10.2x5.8x7	6.35x12x11
Ruido ambiental	Si	Si	No
Forma de construcción del encoder	Disco con franjas blancas y negras.	Disco con franjas blancas y negras	Disco con agujeros.
Aplicaciones	Propósito general	Encoder para ejes, Detección de material reflectante, Propósito general	Manejo de materiales, empaquetado, tacómetros.
Precio	1,50\$	2,50\$	1\$

Fuente: ¹ (TecBolivia, 2013), ² (Geek Factory, 2017), ³ (Carrod Electrónica, 2017)

Dadas las características del entorno de trabajo, se elige el sensor de tipo herradura H20104 porque tiene mayor inmunidad al ruido ambiental provocado por la luz que pueden existir en el campo de maíz y el disco ranurado que utiliza se ve menos afectado por factores como polvo o iluminación que los sensores optoreflexivos, por lo que se dará su operación en escenario previsto.

4.1.6 Selección de las baterías

a. Batería principal

Uno de los aspectos fundamentales a la hora de desarrollar un robot móvil, es su autonomía, pues los escenarios donde este tipo de robot desarrolla su trabajo son donde no hay una red de energía eléctrica presente. En vista de esto, se opta por dotar a los robots móviles de baterías y complementariamente de paneles solares (su uso todavía no está muy extendido debido al alto costo de estos componentes) para que el prototipo cumpla su objetivo de diseño durante el tiempo que pueda otorgar la capacidad de la batería. Por ello, la elección de unas baterías que tengan una gran capacidad de almacenamiento de energía es de vital

importancia para el desarrollo del proyecto, para ello primero se debe conocer el consumo de energía del prototipo, cómo se muestra en la siguiente tabla:

Tabla 19
Consumo energético del prototipo

Cargas	Consumo de corriente(A)	Cantidad	Consumo
Motores principales	20.5	2	41 A
Motor herramienta	3	1	3 A
Servomotores	1	3	3 A
Tarjetas de control y sensores	2.5	1	2.5 A
Luces pilotos	0.3	2	0.6 A
Consumo Neto			50.1 A
Total incluido el 30% por pérdidas de las baterías.			65 A

La batería debe ser capaz de proveer la energía necesaria para abastecer tanto a los actuadores (motores y servomotores) como a la parte electrónica (tarjetas de control, circuitos impresos), por lo que se puede seleccionar dos baterías de 12V a 20 Ah de la marca Sunbright Power AGM, conectadas en paralelo, con lo que proporcionará al prototipo una capacidad total de 40Ah para un tiempo de trabajo estimado de 37 minutos.

Otro punto importante a considerar es el peso de la batería, ya que este ayudará al prototipo que tenga un mayor contacto contra el suelo y ayude a bajar su centro de masa mejorando la estabilidad total del sistema. La batería se puede observar en la figura 104.



Figura 104 Batería sbb AGM 20Ah / 12VDC

Fuente: (Sunbright Power co., 2016)

Las características técnicas de la batería se presentan en la siguiente tabla:

Tabla 20
Características de la Batería sbb AGM 20Ah / 12VDC

Batería SBB AGM 20Ah	
Voltaje	12V
Capacidad	20Ah
Peso	5.5 Kg
Dimensiones	181x77x167 mm
Ciclos	1200 ciclos a 30% DOD
Vida útil	3- 5 Años

Fuente: (Sunbright Power co., 2016)

b. Batería externa para Raspberry Pi

Una vez definido la alimentación principal del sistema eléctrico y electrónico, es necesario considerar una alimentación adicional para el Raspberry Pi, esto se debe a que esta microcomputadora debe seguir encendida aun cuando la batería principal se haya agotado y a la vez para dotarla de una fuente adecuada para sus características de operación. Aprovechando la conexión

MicroUSB que posee el Raspberry Pi, se opta por utilizar una batería externa. Se debe considerar que el sistema embebido también debe alimentar la cámara, por lo que se eligió una batería externa ARGOMTech de 10000mAh, 5V y salida 2 A para alimentar al dispositivo durante 4 horas de uso.



Figura 105 Batería Externa ARGOMTech 10000mAh

Fuente: (ARGOMTech, 2017)

4.1.7 Selección del sistema de comunicación

Una vez elegido los componentes principales de control, se puede decidir ya un sistema que sirva para intercomunicar estos elementos entre sí de manera segura.

La comunicación es parte fundamental del prototipo pues permitirá informar los resultados del proceso de visión al controlador encargado del robot y así mismo conocer el estado de la operación del robot al microcomputador, con lo que se logrará que el prototipo cumpla con su cometido. Así como en otras veces, se analiza los distintos sistemas de comunicación existentes para este tipo de dispositivos, que se encuentran en la tabla 21.

Tabla 21
Protocolos de comunicación para los elementos de control

Característica	Puerto Serie	I2C	SPI
Compatibilidad	Raspberry Pi/Arduino	Raspberry Pi/Arduino	Raspberry Pi/Arduino
Velocidad máxima(bauds) ³	115 200	1 000 000	32 000 000
Tipo comunicación	Half Duplex	Half Duplex	Full Duplex
Modos de comunicación	Punto - Punto	Maestro - Esclavo	Maestro - Esclavo
Pines usados en Raspberry Pi ²	2 o conector USB	4	5
Pines usados en Arduino ¹	2 o conector USB	3	5

Fuente: ¹ (Arduino, 2017), ² (Pinout XYZ, 2017), ³ (Henderson, 2017)

Cómo el tipo de comunicación requerida para el prototipo no es muy compleja pues necesita integrar a dos dispositivos y además aprovechando la interfaz USB para la comunicación de los mismos, se elige la Comunicación Serie (también conocida como UART) a la velocidad máxima 115200 bauds. Además el cable USB blindado que viene con el microcontrolador Arduino asegura la máxima tasa transferencia de datos con mínima presencia de ruido.

4.1.8 Selección del software

Para que cada componente pueda realizar una función, es necesario que cuente con un software específico para programar sus instrucciones. Al tener algunos componentes diferentes, es necesario elegir algunas librerías que permitan completar el programa total del prototipo. Entonces, se empieza por el microcontrolador Arduino, este posee su IDE propio para desarrollo y programación del mismo en C, entonces este es el elegido también por sus múltiples librerías disponibles.

Para el Raspberry Pi, se tiene tres casos, una librería para comunicación y manejo de entradas/salidas digitales, otra para Visión Artificial y luego un framework para desarrollar modelos de machine learning, así para el primer caso se usa la librería WiringPi de (Henderson, 2017), para el procesamiento de imágenes se usa el software libre OpenCV y para el modelo de machine learning se usará Caffe(Las razones para escogerlo se detallan en la siguiente subsección). Se programará a este dispositivo en C++ y como IDE se elige a QT Creator, a través de estas librerías elegidas se desarrolla el control del prototipo.

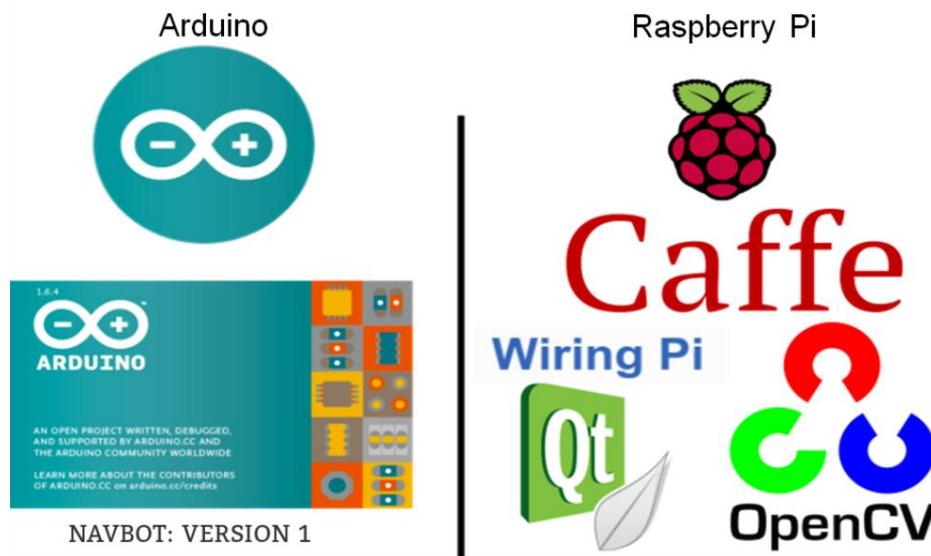


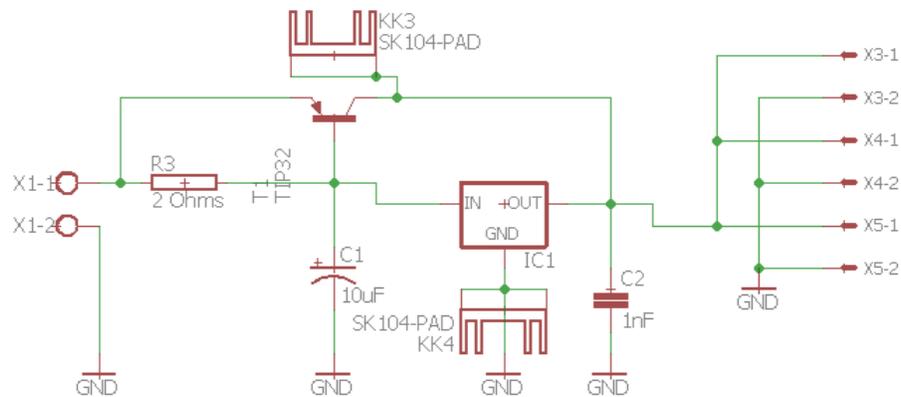
Figura 106 Librerías usadas para cada dispositivo

4.1.9 Circuitos impresos adicionales y otros componentes

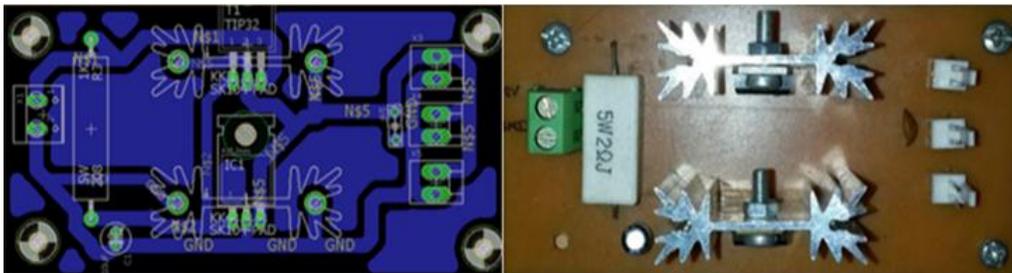
Debido que se tiene algunos componentes electrónicos que deben ser acondicionados e integrados adecuadamente al sistema es necesario el diseño y fabricación de placas electrónicas. Para lo cual se utilizó el software EAGLE, que es una poderosa herramienta de diseño de circuitos impresos (PCB). Se realizaron un total de 4 placas que se describen a continuación.

a. Fuente de poder para los servomotores

Para el diseño de esta fuente se consideró que cada servomotor necesita de una alimentación de 6V a 1 A para proporcionar su máximo torque, por lo que la fuente debe proveer un máximo de seguridad de 6V a 3 A. Para lo cual se utiliza un regulador de voltaje LM7806, un transistor de potencia TIP 42C, una resistencia de potencia de 0.1 Ohms, un capacitor electrolítico de 10 uF, un capacitor cerámico 0.1uF, dos disipadores de calor y conectores tipo polarizados para asegurar unas adecuadas conexiones. Este circuito se muestra en la figura 107.



(a) Esquemático



(b) Diseño

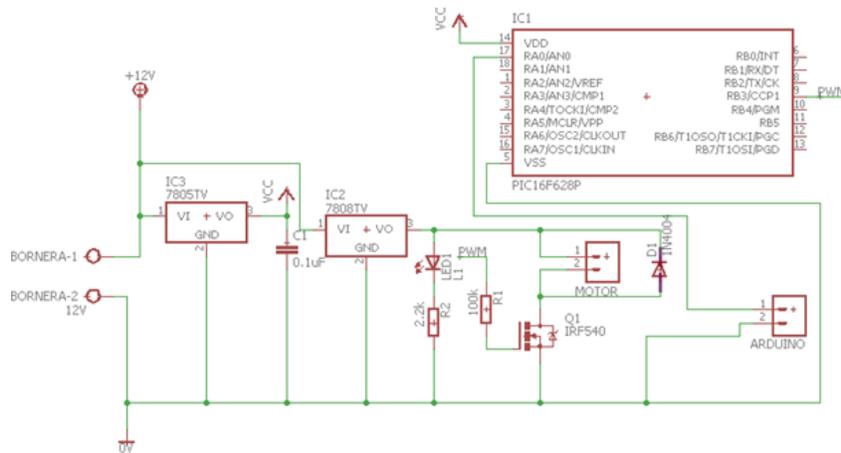
(c) Placa Terminada

Figura 107 Fuente de 6V 3A

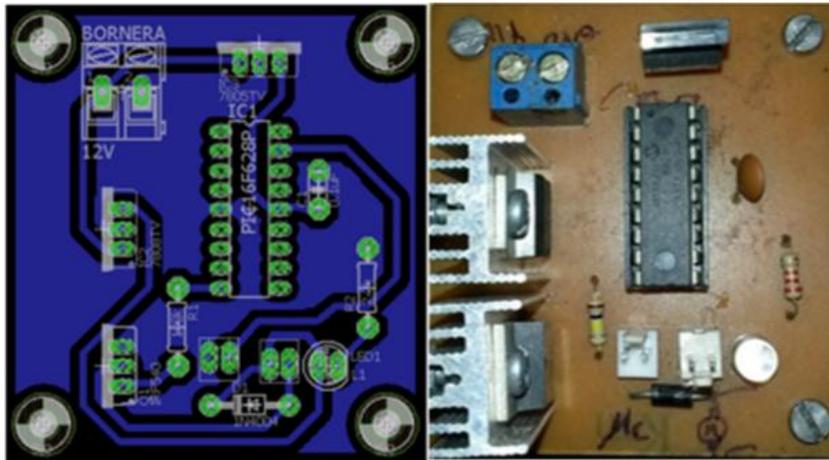
b. Circuito para la herramienta del robot Delta

La herramienta del prototipo de deshierba se pensó para trabajar a dos velocidades, una baja la cual se aplicará al no realizar su función para conservar la

inerencia del movimiento y una velocidad alta para realizar las labores de deshierba. Además el controlador del motor tiene una función de apagado por seguridad cuando inicie o termine su funcionamiento el robot, el diseño del PCB cuenta con microcontrolador PIC16F628A para el control de velocidad, un interruptor de estado sólido MOSFET IRF640 para el manejo del motor, además de reguladores de voltaje, conectores polarizados y un diodo LED para mostrar el funcionamiento del motor. Los componentes y conexiones se pueden ver la figura 108.



(a)Esquemático



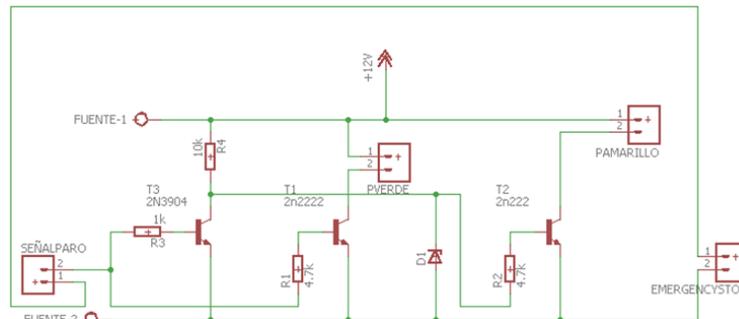
(b)Diseño

(c) Placa Terminada

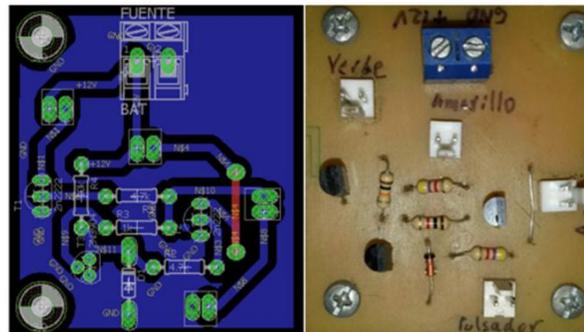
Figura 108 Circuito de control para la herramienta.

c. Circuito para los elementos de control.

Esta placa es auxiliar porque la misma tiene como función de recibir e interpretar las señales del Arduino para activar y desactivar las luces piloto así como enviar al Arduino la señal del paro de emergencia, además para optimizar pines de señal se realiza una compuerta NOT con transistores que se ocupa de la activación de una sola luz piloto a la vez.



(a) Esquemático



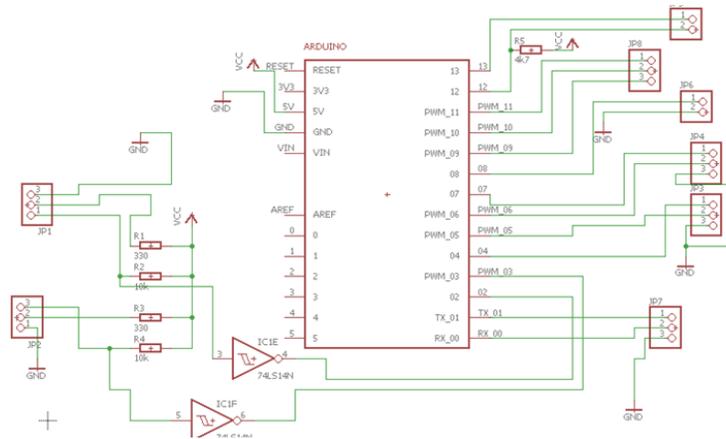
(b) Diseño

(c) Placa Terminada

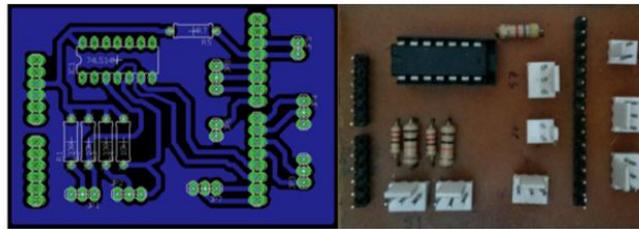
Figura 109 Circuito para elementos de control

d. Shield compatible con Arduino Uno

Esta placa se diseñó con el fin para integrar los distintos periféricos a ser controlados a través del Arduino. Debido a que los sensores para los encoders necesitaban señales digitales puras, la señal es acondicionada con resistencias y un integrado Schmitt Trigger para mejorar el tiempo de conmutación de los pulsos. El circuito se puede ver con mejor detalle en la figura 110.



(a) Esquemático



(b) Diseño

(c) Placa Terminada

Figura 110 Circuito de Shield para Arduino

Para manejar las señales de los motores y servomotores se optó por utilizar conectores polarizados, que permitan establecer con seguridad las conexiones las conexiones evitando fallos debido a las vibraciones a las que está sujeto el robot. Una vez finalizado el diseño, se integra todos los circuitos, que quedarán como el esquemático mostrado en la figura 111:

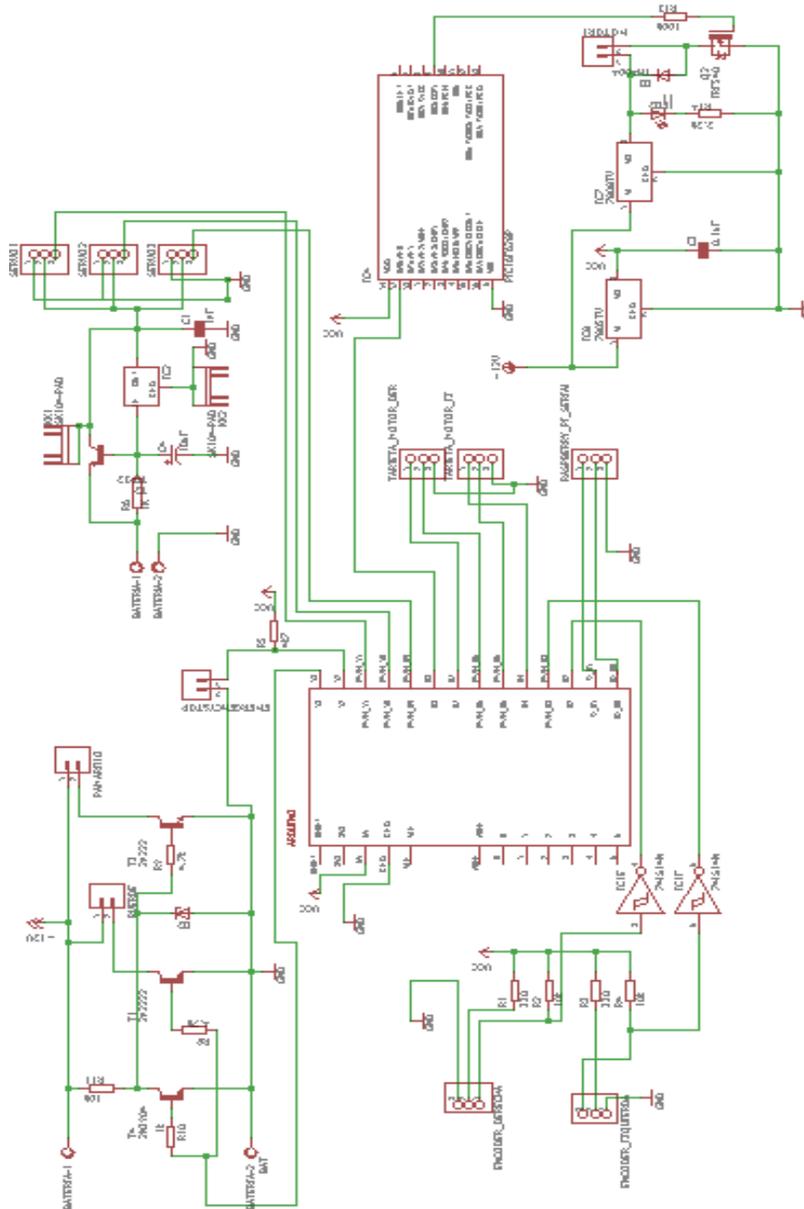
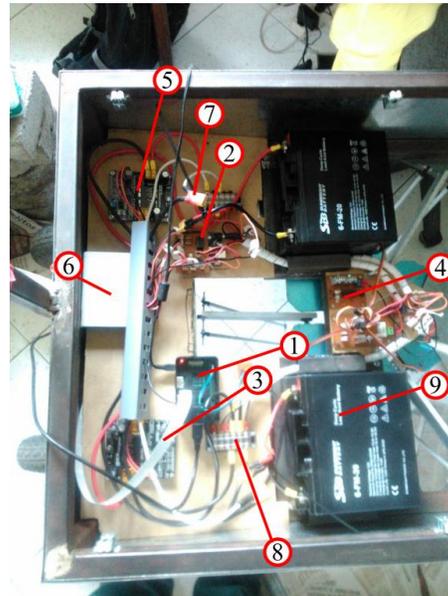


Figura 111 Esquemático del circuito final

e. Conexiones

Para asegurar las conexiones, cómo se mencionó anteriormente, se usó conectores polarizados con cable flexible N°20 y además para conectar la fuente de energía principal, se ocuparon conectores XT60 (que a su vez funcionan como interruptor) con su respectivo cable N°10, el cableado interno fue asegurado con correas y organizador de cables en espiral. Además se ocupó dos borneras para

asegurar la alimentación de los elementos de potencia, entonces el circuito interno del robot, quedó de la siguiente manera (ver figura 112):



1. Raspberry Pi
2. Arduino + Shield
3. Cable CSI
4. PCBs adicionales
5. Cytron MD30C
6. ARGOMTech
7. XT60
8. Bornera
9. Batería 20Ah

Figura 112 Disposición final de los componentes

4.2 Diseño del controlador para la detección de mala hierba

4.2.1 Selección del modelo de machine learning

Uno de los retos más complicados para el proyecto, yace en la clasificación entre las plantas benéficas (maíz) y la maleza en un terreno de cultivo, los avances en sistemas de aprendizaje por máquina como las redes neuronales artificiales (ANN), máquinas de vectores de soporte (SVM) y recientemente las redes neuronales convolucionales (CNN) han permitido la creación de sistemas inteligentes con una precisión cercana a la humana. Su importancia radica en que estos sistemas pueden agregar una dimensión de movilidad para conjuntos robóticos y la liberación de tareas monótonas para el ser humano como en el caso del reconocimiento de caracteres(OCR), entre otras de interés comercial como Big Data (Smola & Vishwanathan, 2008).

Para el reconocimiento de las plantas benéficas de cultivo se han usado distintos tipos de modelos de machine learning, los autores (Tellaeché, Pajares, Burgos-Artizzu, & Ribeiro, 2011) utilizaron SVM para distinguir entre las plantas del cultivo y malas hierbas, (Potena, Nardi, & Pretto, 2016) usaron dos arquitecturas de CNN para realizar la identificación entre el suelo, remolacha de azúcar y maleza, otro modelo de Machine Learning utilizado fue una ANN para la identificación de malezas, este es el caso de (Hong, Lei, & Heping, 2011) quienes aprovechan este modelo para detectar plantas no benéficas bajo un sistema de visión artificial. Con el fin de elegir el modelo de Machine Learning más adecuado para este proyecto, se procedió con el análisis de la tabla 22.

Tabla 22
Comparativa entre distintos modelos de Machine Learning

Tipo ML	SVM ¹	Redes Neuronales ²	CNN ³
Aplicaciones	Clasificación y Regresión	Clasificación, predicción, optimización, Control	Reconocimiento de Imágenes, Procesamiento Natural de voz
Entrenamiento	Supervisado y no supervisado	Supervisado y no Supervisado	Supervisado y no Supervisado
Modelo de funcionamiento	Hiperplanos	Producto, interconexión y funciones de activación	Similar a la corteza visual de los animales
Acierto detección Malas Hierbas(max)	-	95.1%	98.7%
Tasa de Error en Dataset MINST*	0.88%	2.8%	0.61%

* (LeCun, Cortes, & Burges, 2017) *Datos promediados de la tasa de error mostrada la comparación de los modelos en el Dataset de MINST de dígitos escritos para cada modelo*
Fuente: ¹ (Ray, 2015), ² (Universidad de Nevada, 2017), ³ (Deshpande, 2017)

La anterior tabla muestra la comparación entre los distintos modelos de aprendizaje por máquina, como se nota, la que tiene mejores resultados incluido en la detección de malas hierbas es las Redes Neuronales Convolucionales. Las CNN en la actualidad han despuntado permitiendo a los desarrolladores crear aplicaciones con precisión cercana a la humana, un criterio que se verá reflejado en este trabajo.

4.2.2 Selección del framework

Como se discutió en la anterior selección, el método vigente en la actualidad es el de las Redes Neuronales Convolucionales; su precisión y robustez permiten que sea aplicado en la clasificación de plantas de maíz y de maleza. Es necesario elegir un framework que pueda ser implementado en un Raspberry Pi para la aplicación de la red y un computador con tarjeta gráfica NVIDIA GTX950M en donde se desarrollará el modelo. El uso de una GPU supone un tiempo menor de entrenamiento y de pruebas, que es una característica esencial para el proceso de desarrollo del programa. Además, el framework debe correr sobre Linux para que se puede implementar sobre el sistema embebido, una comparación de los softwares más conocidos son presentados a continuación:

Tabla 23
Comparativa de Frameworks de Software Libre

Framework	Caffe ¹	OverFeat ¹	Torch ¹	Tensorflow ²	Theano ¹
Licencia	BSD	Sin especificar	BSD	Apache 2.0	BSD
Lenguaje de programación base	C++	Lua	Lua	Python	Python
Lenguajes de programación compatibles	Python, Matlab	C++, Python	-	Java, C++	-

CONTINÚA 

Compatibilidad con GPU	Si	No	Si	Si	Si
Entrenamiento	Si	No	Si	Si	Si
Modelos pre-entrenados	Si	Si	No	No	No
Compatibilidad Linux	Si	Si	Si	Si	Si

Fuente: ¹ (Jia, y otros, 2014) , ² (Abadi, y otros, 2015)

Por la facilidad de aprendizaje y de implementación, el framework escogido es Caffe. Entre las ventajas que se puede destacar de este software es que el desarrollo de la red se puede hacer a través de un simple archivo Protobuf(Archivo plano de texto) que evita la complejidad de código. Otra de las características importantes es que permite reanudar un entrenamiento cuando se ha interrumpido a fuerza, fácil cambio entre GPU y CPU para pruebas y entrenamiento.

Caffe, además incluye una multitud de herramientas adicionales para generar archivos lmbd(bases de datos para entrenamiento), extracción del promedio de color de las imágenes, con un catálogo de capas bien documentado además de múltiples solucionadores para el entrenamiento de red y para graficar las fases del entrenamiento.

4.2.3 Preparación de la red

a. Dataset

Para lograr el objetivo de entrenar una red robusta que pueda diferenciar acertadamente el maíz de la maleza se ha utilizado un dataset con 861 imágenes de la clase mala hierba y con 2835 imágenes de la clase maíz procesadas con su fondo segmentado y etiquetado manualmente. De esta manera se asegura que la red aprenda las características de forma y color que tiene una planta de maíz y

una planta de mala hierba. El proceso se ilustra de una mejor forma en el siguiente diagrama de bloques.

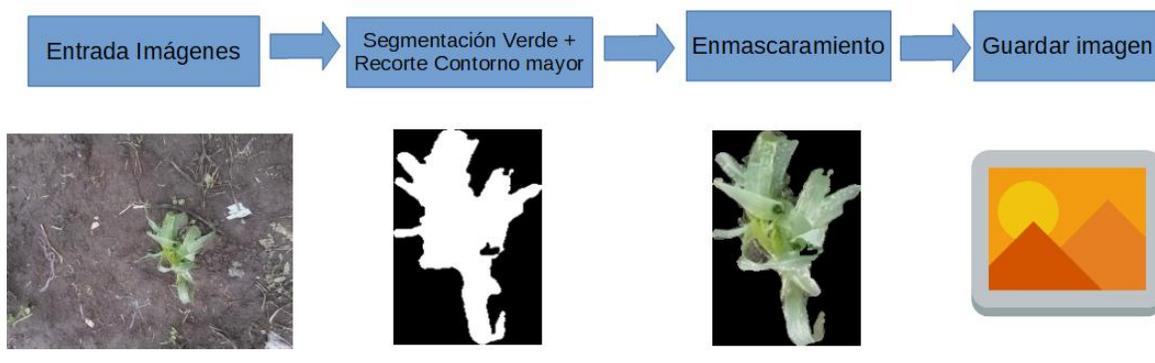


Figura 113 Proceso de Segmentación de las Muestras

Uno de los problemas encontrados es que este tipo de redes necesita un número grande de imágenes para un entrenamiento adecuado, pero (Lee, Chan, Mayo, & Remagnino, 2017) menciona que el dataset puede extenderse al girar las imágenes. Entonces se giran las imágenes cada 30 grados, con lo que el arreglo final de imágenes queda con 10332 imágenes de la clase mala hierba y 34222 imágenes de la clase maíz.

De las dos clases de imágenes se toma la quinta parte de las imágenes para la valoración de la red (en un orden totalmente aleatorio para cerciorarse que la red se entrene y se valide con cualquier imagen en el dataset). Al final se forman dos datasets, que quedan compuestos de la siguiente forma: Para la clase mala hierba 8266 imágenes de entrenamiento y 2066 imágenes de validación; para la clase maíz 27378 imágenes de entrenamiento y 6844 imágenes de validación. Entonces se forman dos datasets y se almacenan en bases de datos de tipo LMDB, uno de entrenamiento y otro de validación.

A su vez es conveniente extraer el promedio de color de las imágenes (traducción textual del término original inglés “image mean”). Este valor permite “centrar” los datos para que en el entrenamiento los gradientes no se salgan fuera de control, Caffe permite realizar este proceso de manera nativa. Debido al fondo

negro, los valores medios de color de la imagen quedaron para cada canal de color de la siguiente forma: Rojo: 43, Verde: 58, Azul: 52.

b. Desarrollo de la red

Con los archivos Imdb generados, se adapta la arquitectura de la red a las imágenes de entrada. En Caffe es necesario generar dos archivos protobuf(ambos archivos tienen extensión prototxt y son representados como archivos planos de texto).

El primer archivo contiene la red modificada para entrenamiento, en la capa de entrada se especifica el archivo promedio de color de imágenes (image mean) y los datasets de entrenamiento y valoración en formato Imdb, la capa de salida contiene una capa SoftmaxwithLoss, esta capa es fundamental para que el solucionador de la red realimente el dato del error en la etapa de entrenamiento.

El otro archivo correspondiente al de desarrollo (del inglés deploy), debe tener en la capa de entrada la imagen a clasificar y en la salida, como ya no es preciso obtener los datos de pérdida de la capa, se sustituye ésta con la capa Softmax que sirve para obtener los valores finales de la clasificación de la red.

Aparte de estos dos parámetros, Caffe permite interactuar con 6 distintos tipos de solucionadores de los parámetros de red (Jia, y otros, 2014) La utilización de los solucionadores permite obtener distintas formas de mejorar la pérdida en la capa de salida y de optimizar de mejor forma el entrenamiento de las capas. La elección del solucionador se realiza de manera empírica pues no hay regla que permita elegir el mejor.

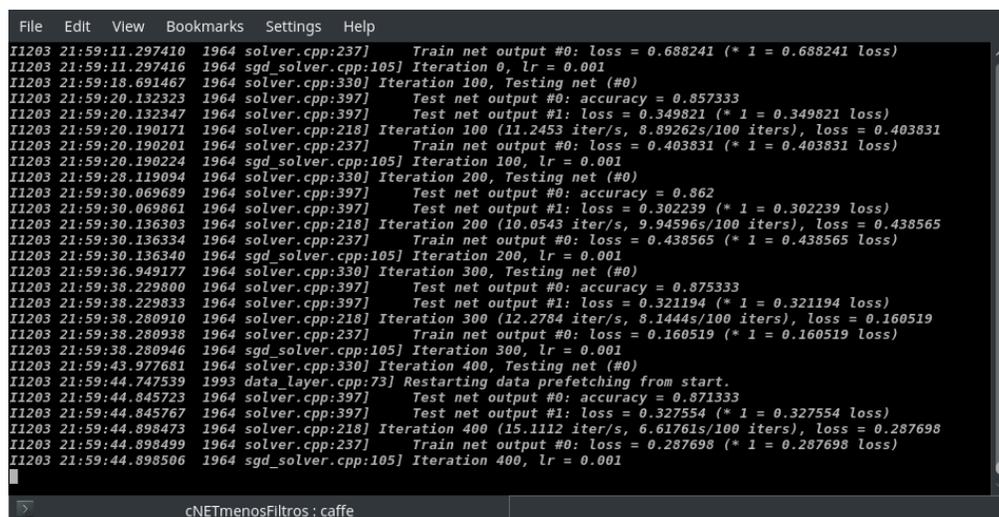
Una vez culminada la preparación de los archivos necesarios, se procede con el entrenamiento de la red. El framework Caffe es muy versátil con herramientas útiles para el entrenamiento y para graficar sus resultados. En la actualidad el entrenamiento es un proceso complejo, que requiere un amplio poder computacional debido a los múltiples parámetros que contienen las redes y las

miles de iteraciones necesarias para asegurarse que el error de la capa de salida sea el mínimo.

c. Entrenamiento

Para el entrenamiento se usó el Sistema Operativo Kubuntu con un computador Core i7 con ocho núcleos y una tarjeta gráfica Nvidia 950M. La ventaja de la tarjeta gráfica según (NVIDIA, 2017) es su computación paralela, su eficiencia en comparación con la CPU y las múltiples aplicaciones como Machine Learning, Visión Artificial, Data Mining, etc. Caffe está ligado con CUDA con lo cual se ofrece una experiencia rápida para el entrenamiento a través del uso de computación por GPU.

Cuando el entrenamiento empieza, la consola devuelve el número de iteraciones, la precisión y la pérdida en la función de salida. Otra de las ventajas de Caffe es la de guardar instantáneas cada cierto intervalo de iteraciones por lo que se puede reanudar si se tuvo que pausar el entrenamiento forzosamente. Cuando finaliza el entrenamiento se genera un archivo .caffemodel y se elige la instantánea que tiene una menor pérdida en su capa de salida. Este modelo será luego añadido al programa y es el que contiene los valores de los parámetros entrenados.



```

File Edit View Bookmarks Settings Help
I1203 21:59:11.297410 1964 solver.cpp:237] Train net output #0: loss = 0.688241 (* 1 = 0.688241 loss)
I1203 21:59:11.297416 1964 sgd_solver.cpp:105] Iteration 0, lr = 0.001
I1203 21:59:18.691467 1964 solver.cpp:330] Iteration 100, Testing net (#0)
I1203 21:59:20.132323 1964 solver.cpp:397] Test net output #0: accuracy = 0.857333
I1203 21:59:20.132347 1964 solver.cpp:397] Test net output #1: loss = 0.349821 (* 1 = 0.349821 loss)
I1203 21:59:20.190171 1964 solver.cpp:218] Iteration 100 (11.2453 iter/s, 8.89262s/100 iters), loss = 0.403831
I1203 21:59:20.190201 1964 solver.cpp:237] Train net output #0: loss = 0.403831 (* 1 = 0.403831 loss)
I1203 21:59:20.190224 1964 sgd_solver.cpp:105] Iteration 100, lr = 0.001
I1203 21:59:28.119894 1964 solver.cpp:330] Iteration 200, Testing net (#0)
I1203 21:59:30.069689 1964 solver.cpp:397] Test net output #0: accuracy = 0.862
I1203 21:59:30.069861 1964 solver.cpp:397] Test net output #1: loss = 0.302239 (* 1 = 0.302239 loss)
I1203 21:59:30.136303 1964 solver.cpp:218] Iteration 200 (10.0543 iter/s, 9.94596s/100 iters), loss = 0.438565
I1203 21:59:30.136334 1964 solver.cpp:237] Train net output #0: loss = 0.438565 (* 1 = 0.438565 loss)
I1203 21:59:30.136340 1964 sgd_solver.cpp:105] Iteration 200, lr = 0.001
I1203 21:59:36.949177 1964 solver.cpp:330] Iteration 300, Testing net (#0)
I1203 21:59:38.229800 1964 solver.cpp:397] Test net output #0: accuracy = 0.875333
I1203 21:59:38.229833 1964 solver.cpp:397] Test net output #1: loss = 0.321194 (* 1 = 0.321194 loss)
I1203 21:59:38.280910 1964 solver.cpp:218] Iteration 300 (12.2784 iter/s, 8.1444s/100 iters), loss = 0.160519
I1203 21:59:38.280938 1964 solver.cpp:237] Train net output #0: loss = 0.160519 (* 1 = 0.160519 loss)
I1203 21:59:38.280946 1964 sgd_solver.cpp:105] Iteration 300, lr = 0.001
I1203 21:59:43.977681 1964 solver.cpp:330] Iteration 400, Testing net (#0)
I1203 21:59:44.747539 1993 data_layer.cpp:73] Restarting data prefetching from start.
I1203 21:59:44.845723 1964 solver.cpp:397] Test net output #0: accuracy = 0.871333
I1203 21:59:44.845767 1964 solver.cpp:397] Test net output #1: loss = 0.327554 (* 1 = 0.327554 loss)
I1203 21:59:44.898473 1964 solver.cpp:218] Iteration 400 (15.1112 iter/s, 6.61761s/100 iters), loss = 0.287698
I1203 21:59:44.898499 1964 solver.cpp:237] Train net output #0: loss = 0.287698 (* 1 = 0.287698 loss)
I1203 21:59:44.898506 1964 sgd_solver.cpp:105] Iteration 400, lr = 0.001
  
```

Figura 114 Proceso de entrenamiento en Linux

4.2.4 Selección y entrenamiento de la arquitectura de red

a. Selección de la red

Las redes neuronales convolucionales pueden representarse con múltiples arquitecturas, muchas de ellas pueden ser simples con pocas capas o complejas como el caso de las redes ganadoras de los retos de clasificación de ImageNet (ImageNet, 2017). Caffe incluye redes de ejemplo en su llamado Caffe Zoo Model que son un excelente recurso para el aprendizaje y la experimentación. Para este caso se consideraron los modelos de Lenet y Alexnet para seleccionar la arquitectura de la red. Además (Potena, Nardi, & Pretto, 2016) mencionan dos arquitecturas adicionales de redes probadas que tienen una alta precisión para identificación entre cultivos y mala hierba: la sNET y cNET, que se adaptaron para el presente trabajo, con el fin de implementar las definiciones al formato de Caffe.

Estas cuatro alternativas de arquitectura fueron entrenadas con un solucionador de tipo AdaDelta, a continuación, se muestra la tabla 24 con los resultados obtenidos para cada tipo de red.

Tabla 24
Comparación técnica de las arquitecturas seleccionadas

Características	Lenet	Alexnet	cNET	sNET
Tamaño de las imágenes de entrada	32x32	64x64	64x64	64x64
Número de Capas	9	11	8	4
Número de Parámetros	652500	20166688	6421568	135872
Número de Iteraciones para entrenamiento	3000	3000	4500	3000
Precisión de la red con imágenes de prueba(Final)	86.5	93.9	96.4	80.4
Función de Pérdida(Final)	32.8	15.3	13.7	15.3

Como se aprecia en la tabla anterior, el mejor resultado con el Dataset de dos clases lo logró la red cNet, esta contiene menos parámetros que la red Alexnet lo que permitirá obtener un menor coste computacional en el sistema embebido. También la función de pérdida es la menor de todos lo que demuestra la verdadera robustez de la red.

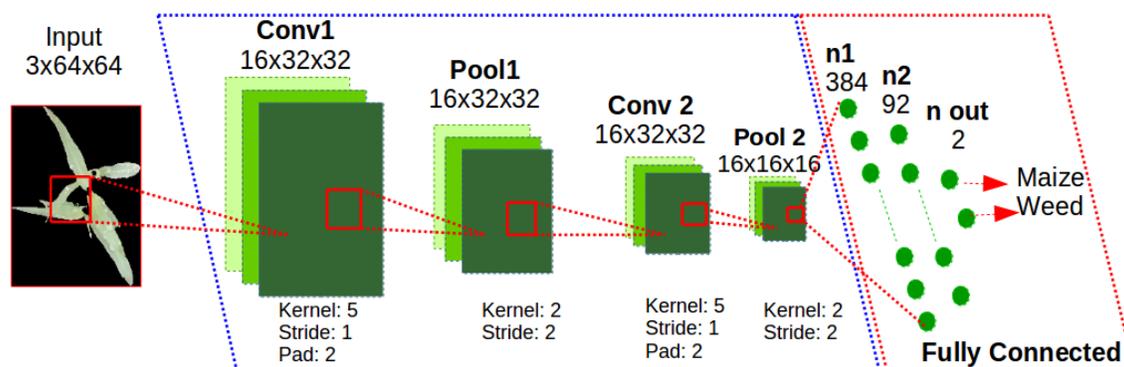


Figura 115 Detalle de la red cNet

b. Selección del solver

Como se mencionó anteriormente, hay distintos tipos de solvers, en la tabla siguiente se presentan resultados de entrenamiento al usar cinco tipos admitidos por Caffe. Cabe destacar que, dependiendo del tipo de solucionador se pueden obtener mejores resultados en la red con lo cual se beneficiaría la precisión del proyecto. Cada solucionador fue probado con la Arquitectura cNET con 16 filtros.

Tabla 25
Comparación del entrenamiento con distintos tipos de solvers

Parámetros	Adam	Auto	AdaDelta	SGD	RMSProp
Número de Iteraciones para entrenamiento	4500	4500	4500	4500	4500
Precisión de la red con imágenes de prueba (Final) (%)	97.2	96.9	93.5	95.5	71.2
Función de Pérdida (Final) (%)	8	7	15	13	60

Como se muestra en la tabla 25, la mayoría de los solucionadores tienen el mismo valor de precisión en la red a excepción del RMS Prop, su diferencia es la función de pérdida. Una menor función de pérdida representa el objetivo de todo solucionador de una red, por lo que se prefieren redes que tengan una menor función de pérdida. Debido a esto, el Solver elegido será el Adam.

En la siguiente figura se presentan los resultados de entrenamiento con el Solver Adam obtenidos con el dataset de prueba ante un número de 5000 iteraciones. Nótese que la precisión de entrenamiento se asemeja a una función exponencial, por lo que la función de pérdida baja constantemente hasta su estabilización.

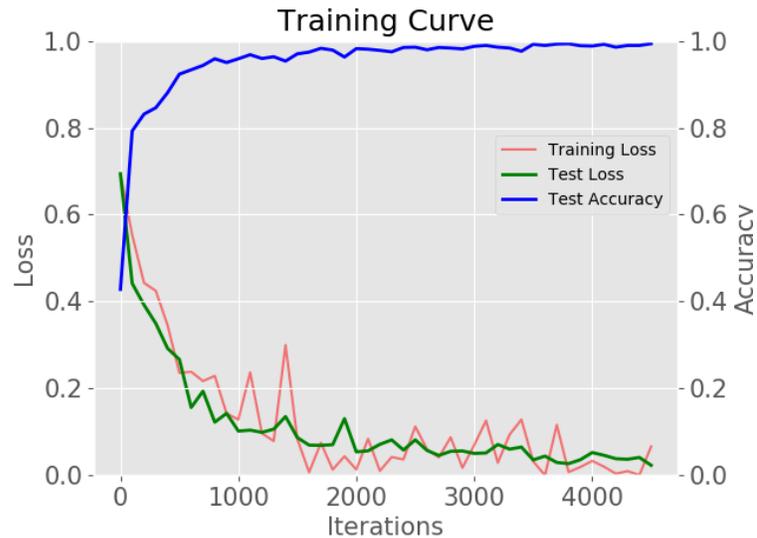


Figura 116 Proceso de entrenamiento de la red usando el Solver Adam

4.2.5 Clasificación

El modelo obtenido al final del entrenamiento, es un binario con extensión caffemodel. Este archivo contiene los parámetros de entrenamiento para cada uno de los elementos de la capa. Caffe incluye extensiones para C++ y Python, pero en este caso se usará C++ porque a diferencia de Python es un lenguaje de programación compilado que correrá directamente sobre el núcleo de Linux y además generará un archivo compilado binario que será usado más adelante para

que en el arranque, el Raspberry Pi inicie directamente con el programa del prototipo. Además C++ tiene la ventaja de poseer gran multitud de librerías, soporte para programación multihilos y librerías de control para el Raspberry Pi.

Cómo Caffe es un framework con extensos ejemplos, se analiza un ejemplo llamado *Classification.cpp* que sirve para clasificar imágenes con modelos ya entrenados, para este caso se adapta al código para que incluya el procesamiento de imágenes y realice la clasificación de las mismas. Para esto se incluyó varias etapas, que se describen a continuación.

a. Adquisición de imágenes

El uso de la cámara CMOS en el software OpenCV es a través de los Drivers de Broadcom especificados en el Linux Kernel. Como OpenCV es compatible con los drivers video4linux (v4l y v4l2), se utilizan los drivers especificados por Raspberry Pi para sus módulos de cámara, el bcm2835-v4l2 (V4L2 BCM2835 Driver, 2017).

Para el proyecto, el uso del driver es estrictamente necesario, por lo que su llamada se ejecuta al inicio del sistema, entonces es necesario incluir el nombre del driver (o también conocido como módulo de kernel) *bcm2835-v4l2* en una línea aparte del archivo */etc/modules*. Después de esto se procede con la interfaz de v4l2 hacia OpenCV, luego el objeto VideoCapture de OpenCV puede obtener imágenes de la cámara que en este caso es de 1280x720 píxeles con el fin de adquirir un buen detalle del área de cultivo, por lo que el diagrama de adquisición de imágenes queda de la siguiente forma:

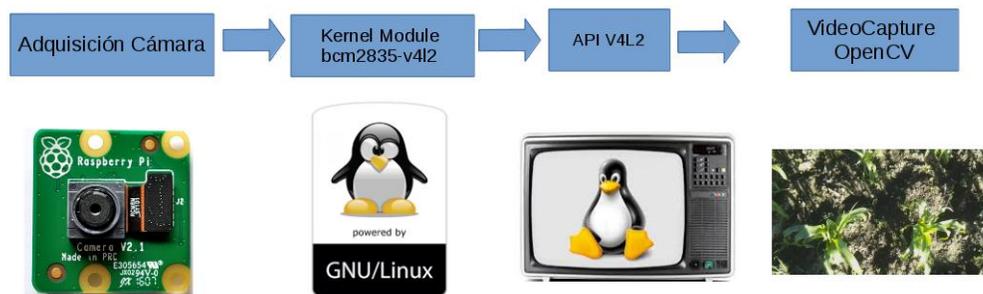


Figura 117 Diagrama de Bloques del Proceso de Adquisición de imágenes

b. Pre-procesamiento de imágenes

Una vez obtenido las imágenes, se necesita analizar cada planta de cierto tamaño dentro del cultivo para determinar si es de maíz o a su vez de mala hierba. Primero se separa las imágenes en capas y luego el canal verde es normalizado para minimizar los posibles efectos que producen la luz y las sombras en el terreno de cultivo. Entonces se procede a aislar el canal verde de la imagen, sustrayéndolo de los canales restantes a través de la siguiente forma descrita por (Wang, Meng, & Mei, 2013).

$$S = 2 * G - R - B \quad (32)$$

Dónde:

G: Canal verde

R: Canal rojo

B: Canal azul

La imagen luego debe ser procesada con un *medianBlur* para filtrar posibles ruidos, entonces se binariza con un umbral binario y de tipo OTSU. Una vez obtenida la imagen binarizada pasa por una dilatación para evitar las posibles malas detecciones por el efecto de la luz sobre la hoja de la planta. Luego la imagen binaria pasa por un proceso de detección de contornos, donde estos se guardan en un vector, el proceso mencionado se ilustra mejor en la figura 118.

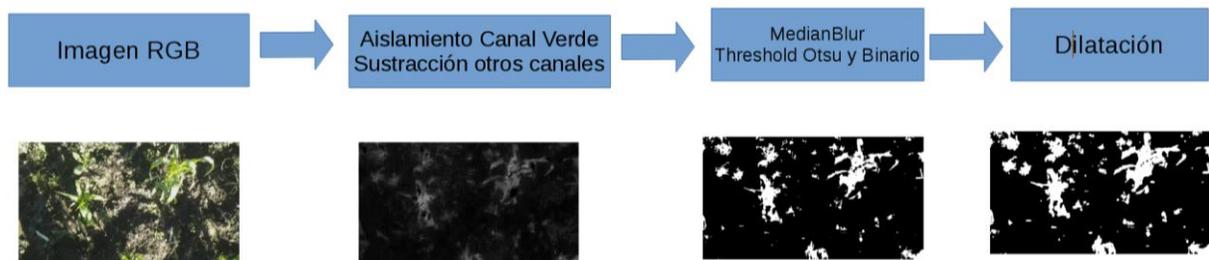


Figura 118 Diagrama de bloques del proceso de Pre-Procesamiento

c. Reconstrucción de contornos

Cómo en el exterior se encuentran muchas veces condiciones extrañas (polvo, luz irregular) que afectan el pre-procesamiento de las imágenes, es necesario definir un proceso adicional, pues se necesita que las imágenes estén lo más completas posible para poder clasificarlas y enviarlas a la siguiente etapa de control, los problemas más comunes encontrados en el cultivo de maíz, son los que se presentan en la siguiente figura:



(a) Por tierra

(b) Por sombra

Figura 119 Error de binarización

Como se observa en la anterior figura, en *a*) las plantas que están más cerca del suelo (comúnmente mala hierba) pierden su color verde por que acumulan partículas de tierra, como se aprecia en *b*), las sombras producidas por las plantas grandes, hacen que ciertas partes de la planta no puedan ser segmentadas, además hay muy pequeñas detecciones que pueden deberse a

pequeñas plantas o incluso a piedras con algún organismo productor de clorofila, por lo que es necesario desarrollar un paso extra para compensar estos efectos, por lo que se plantea el siguiente algoritmo.

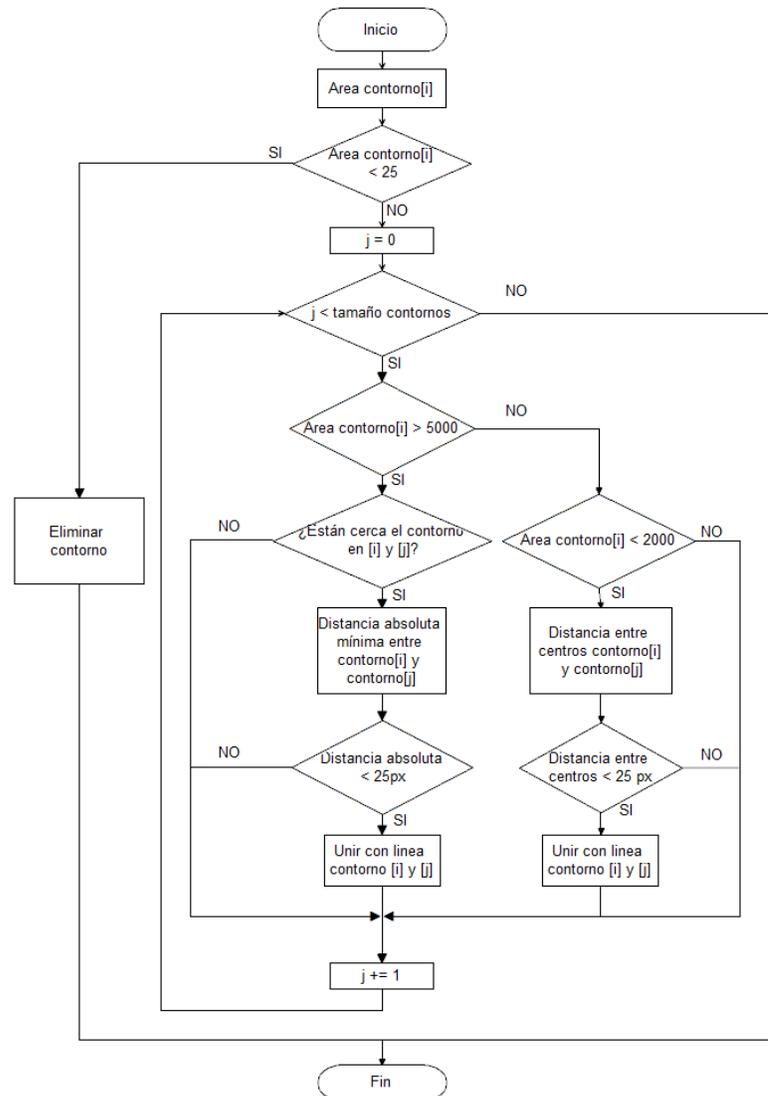


Figura 120 Diagrama de flujo la reconstrucción de contornos

Luego de implementar el algoritmo para cada contorno, se puede apreciar que los resultados de procesamiento son aptos para ingresar a la red y además se tendrá menos puntos que enviar al robot Delta, con lo que se mejora los tiempos

de trabajo del prototipo, entonces un ejemplo de los resultados se presenta a continuación:



Figura 121 Contornos reconstruidos para los casos anteriores

d. Segmentación

Una vez mejorada la imagen binaria, es necesario analizar cada uno de los contornos, entonces se encontrará que algunas plantas son fácilmente identificables como mala hierba. Los contornos menores al tamaño de la capa de entrada de la red son considerados como mala hierba, pues su tamaño es mucho menor a maíz, especialmente en sus etapas iniciales. Para una mejor detección de las imágenes, se enmascara los contornos con el fin de tener más características que pueda analizar la red, después de eso se recorta para obtener la región de interés (ROI) final de la imagen, que será el dato que entre a la red neuronal convolucional para su análisis, por lo que el procesamiento queda de la siguiente forma:

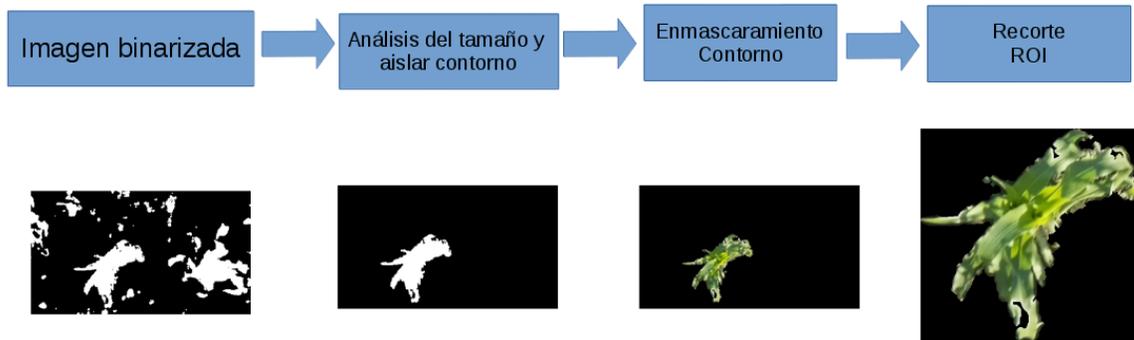


Figura 122 Diagrama de Bloques de la Etapa Segmentación

e. Clasificación

Una vez obtenido el ROI, las imágenes son analizadas para clasificarlas dentro de las dos clases definidas: Clase mala hierba y clase maíz. Para ello Caffe posee funciones para llevar la variable Mat de OpenCV (que es la matriz de 3 capas que contiene la imagen RGB) hacia una entrada de tipo 'Blob' necesaria para la clasificación en el framework, luego se obtienen dos probabilidades de la pertenencia a cada clase, por lo que se elige la clase que tenga la mayor probabilidad de la clasificación. Este proceso se muestra de mejor manera en el gráfico siguiente:

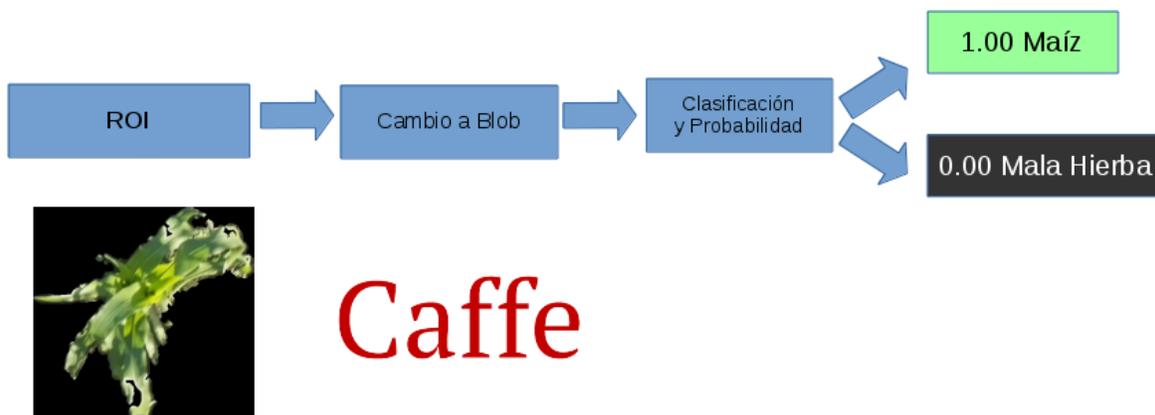


Figura 123 Diagrama de procesos de clasificación con una planta de maíz

Para almacenar los datos de la clasificación, se crea una estructura de datos que almacenará la información que sea de utilidad para las siguientes etapas del prototipo. La información más importante para este tipo de dato son: el punto que representa la posición en x y y de la esquina superior izquierda del rectángulo que alberga al contorno clasificado, el ancho que es el la longitud en x desde el origen de coordenadas del rectángulo hasta el punto extremo en x del mismo y el alto que es la longitud desde el punto de origen de coordenadas del rectángulo hasta el último punto en y del mismo.

Para separar a todos los elementos clasificados se generan dos vectores, uno por cada clase de la estructura *objetored* con los nombres *maicestotales* y *malashierbastotales*, lo explicado en este párrafo se puede apreciar mejor en la siguiente imagen.

```
struct objetored{
    cv::Point punto;
    int area;
    float probabilidad;
    string clase;
    int ancho;
    int largo;
};
std::vector<objetored> maicestotales;
std::vector<objetored> malashierbastotales;
```

Figura 124 Estructura y vectores para almacenar los datos de clasificación

Una vez que el proceso de clasificación ha finalizado, se utilizan los datos para el siguiente proceso que es el establecimiento de la trayectoria.

4.3 Diseño del controlador para establecimiento de la trayectoria

Una vez que el sistema de clasificación ha sido establecido, es necesario guiar al prototipo de robot móvil a través de la hilera del maíz, como es bien conocido, el desarrollo del maíz no es tan regular e incluso es impredecible y

dichas irregularidades deben ser compensadas en la trayectoria de la plataforma móvil con el fin de no afectar a las plantas beneficiosas que se quiere resguardar, para ello se presenta el siguiente desarrollo:

4.3.1 Generación de hileras

Esta parte no es más que establecer en qué posición se encuentran las plantas de maíz clasificadas al lado izquierdo y derecho de la imagen de la cámara, con estos datos se puede encontrar el centro de la hilera y mandar a corregir la posición de la plataforma móvil a través del navegador, los datos obtenidos para esta etapa son los del vector *maicestotales* pues las plantas de maíz clasificadas son los únicos elementos que intervendrán en la generación de las hileras. El proceso de generación de hilera cuenta con los siguientes subprocesos:

a. Encontrar los puntos extremos

El primer paso para identificar una hilera es conocer las posiciones extremas de las plantas de maíz. Como la distribución de las plantas es casi regular, con la información del rectángulo que contiene los contornos clasificados como maíz en el vector *maicestotales* se puede diferenciar fácilmente la planta que está más cercana al borde izquierdo y derecho de la imagen en el eje x, después con los datos máximos se saca un promedio para encontrar un falso centro, así las plantas que estén a la izquierda del falso centro corresponderán a la hilera izquierda y las que no, a la hilera derecha. Este proceso se muestra de mejor forma en el siguiente diagrama de flujo mostrado en la figura 125.

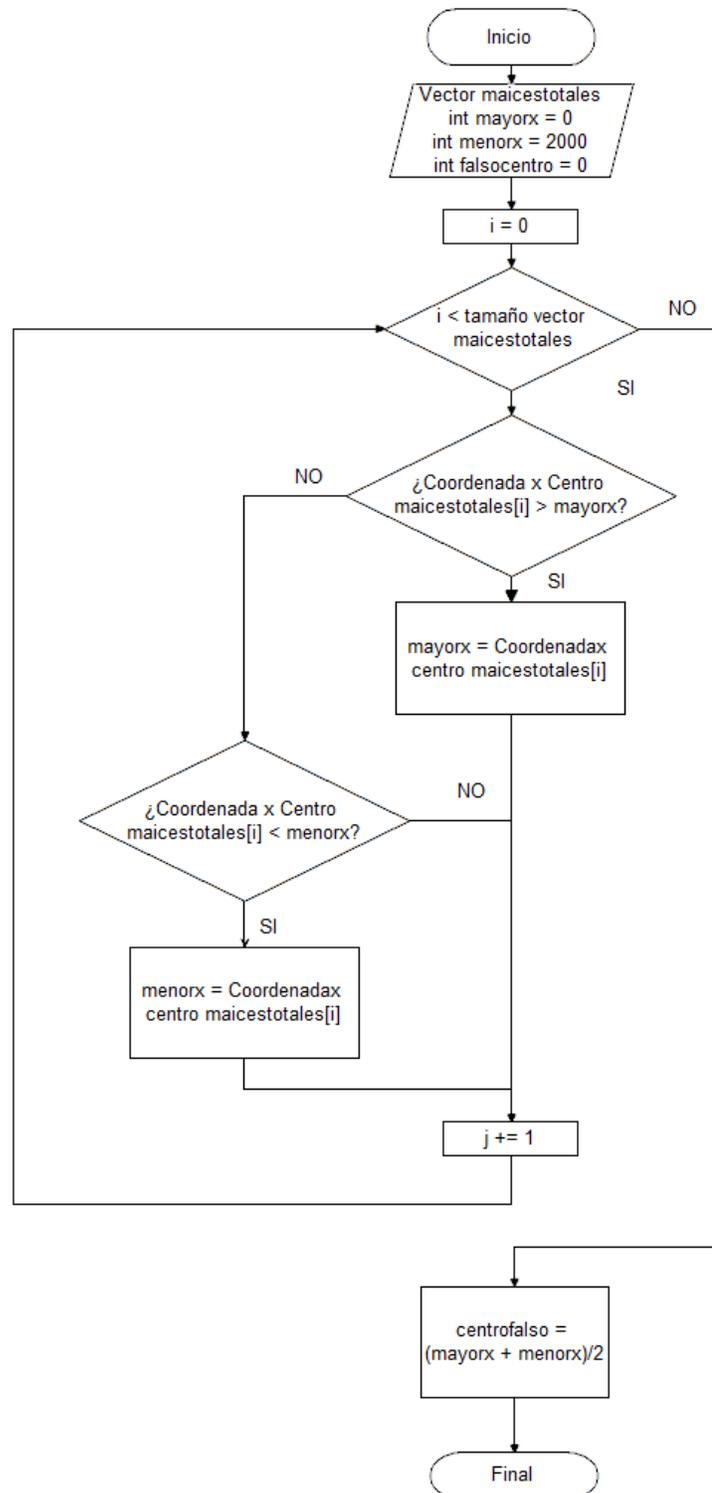


Figura 125 Diagrama de flujo de búsqueda de extremos



Figura 126 Ejemplo de la implementación del algoritmo anterior

Como se nota en la anterior figura, cuando se encuentra el falso centro es más fácil discriminar los elementos que pertenecen a cada hilera. Los datos discriminados se guardan en vectores independientes para su posterior procesamiento. Como el proceso puede variar, se presenta una solución para el posible error de detección.

b. Diferenciar entre una o dos hileras

Cómo el prototipo es autónomo, también surgen ciertos problemas cuando éste encuentre que los puntos extremos que se obtuvieron anteriormente estén muy cerca el uno del otro, esto es debido a que el robot o la distribución de las plantas hacen que sólo se muestre una hilera reconocible en la imagen, entonces se presentan dos casos para generar la hilera, cuando haya dos hileras reconocibles o una sola, entonces para asegurarse que haya dos hileras se divide el valor *mayorx*/*menorx* (siempre que *menorx* sea mayor a 0), si este valor es efectivamente mayor a 1.7 entonces se trata de dos hileras, caso contrario, una sola. Se eligió 1.7 porque al menos se debe tener una relación de una planta y media de distancia adicional para considerar que se detectó dos hileras. En la siguiente figura se puede notar un caso cuando se reconoce solo una hilera, en este caso es por una imagen borrosa a causa del movimiento.



Figura 127 Detalle del algoritmo cuando sólo hay una hilera visible

Ahora, para calcular el centro de la hilera, es necesario considerar los dos casos mencionados anteriormente, como se muestra en la siguiente sección.

c. Cálculo del centro de la hilera

Conocidos los dos casos que pueden surgir para calcular el centro de la hilera, se utiliza los datos que se obtienen en el proceso anterior, entonces se genera código para cada caso, en el caso de que se detecten dos hileras, solo cabe encontrar el promedio de las posiciones en x de los elementos que se encuentran en el vector de la hilera izquierda y realizar el mismo procedimiento a los elementos de la otra hilera, por lo que queda realizar un último promedio entre estos dos datos para establecer el centro de la hilera, como se muestra en la siguiente figura.



Figura 128 Cálculo del centro en dos hileras (Círculo negro)

Para el caso donde se detecte una sola hilera, igual que el paso anterior se realiza un promedio de las posiciones en x de los elementos del vector, luego que se obtenga el valor promediado, discriminar si este valor se encuentra a la derecha o a la izquierda de la imagen, entonces se promedia para el primer caso con 1280 y para el otro caso, se promedia con cero, por lo tanto el robot podrá corregir la posición hasta la siguiente captura de imagen. Un ejemplo de este procesamiento se muestra a continuación.



Figura 129 Centro para una única hilera (Círculo negro)

Cómo se aprecia en la anterior figura, el programa detectó una hilera a la derecha, por lo que el centro de la hilera estará distribuida a la izquierda, para así lograr que la plataforma móvil corrija este inusual movimiento. Una vez finalizado este proceso, se debe pasar a la transmisión de datos hacia la plataforma móvil, que es el proceso que se describe a continuación.

4.3.2 Transmisión de los datos

a. Acondicionamiento de datos

Como los datos obtenidos en la generación de hilera son los datos correspondientes a las posiciones de los píxeles de la imagen, falta realizar una conversión que permita trasladar estos datos hacia la longitud en el campo de

trabajo, para ello simplemente se realiza una relación como se muestra en la siguiente ecuación.

$$\frac{\textit{posición en pixeles}}{\textit{ancho imagen en pixeles}} = \frac{\textit{posición en mm}}{\textit{ancho campovisión en mm}} \quad (33)$$

$$\textit{posición en mm} = \frac{\textit{ancho campovisión en mm} * \textit{posición en pixeles}}{\textit{ancho imagen en pixeles}} \quad (34)$$

Como se conoce que el ancho de campo de visión es de 1525mm y el ancho de la imagen es de 1280px, entonces la relación se puede simplificar a:

$$\textit{posición en mm} = \frac{1525\textit{mm} * \textit{posición en pixeles}}{1280\textit{px}} = 1.19 * \textit{posición en pixeles}$$

Entonces, está relación es válida para transformar de pixeles a milímetros en todos los procedimientos que sean necesarios.

b. Envío de datos

Con los datos obtenidos, se procede a enviarlos al Arduino para que se puedan dar paso al siguiente proceso que es la navegación, cómo se señaló en la sección 4.1.7. El tipo de comunicación es serial para el uso a una velocidad de 115200 bauds para tener poco tiempo de espera entre la comunicación de los datos, para establecer la comunicación del Raspberry Pi, se utilizó la librería WiringPi de (Henderson, 2017) y para comunicar los datos se utilizó el siguiente procedimiento:

- Enviar desde el microcomputador el carácter 'o' para comenzar la transmisión
- Si recibe el Arduino el carácter, enviar la 'a' para confirmar la comunicación
- Enviar desde el Raspberry Pi en forma de un vector de caracteres de cuatro cifras la coordenada transformada a mm del centro de la hilera
- Enviar desde el Arduino una confirmación con el carácter 'f'

El proceso de transmisión, se ilustra mejor en la figura 130:

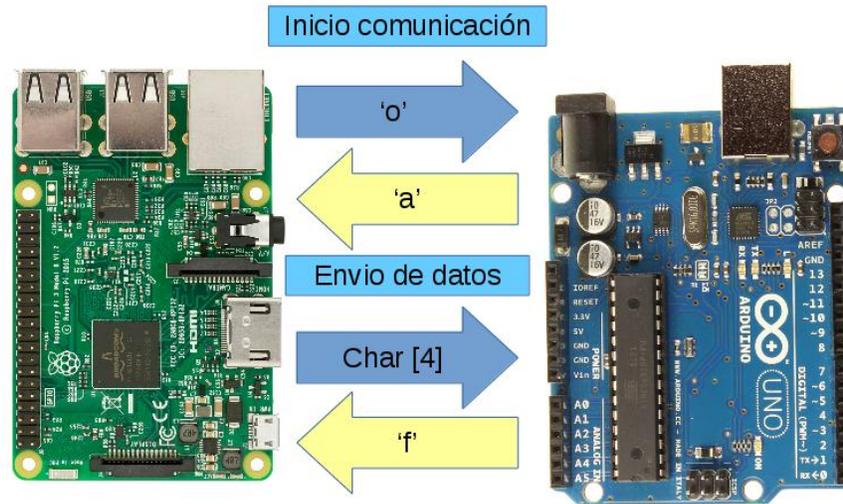


Figura 130 Diagrama de transmisión del centro de la hilera

4.3.3 Control de localización.

La localización del robot es una de las partes principales del prototipo, pues esta técnica permite al mismo el control sobre la trayectoria a seguir dentro del cultivo, por ello, para realizar el control de la navegación, se aplica la técnica de dead reckoning llamada odometría, la cual mediante cálculos de acuerdo a la configuración del robot permite saber las coordenadas del prototipo y su ángulo de orientación con los datos fruto de la lectura de los encoders de las ruedas.

a. Odometría

Para esta técnica se requiere identificar la configuración del robot (que en este caso es de tipo diferencial). Para obtener la localización del prototipo, primero se debe medir la distancia entre las ruedas, el radio de las mismas y conocer la resolución de los encoders, el desarrollo de las ecuaciones permiten obtener los parámetros de salida: Posición(x,y) y orientación(θ) que serán los que se muestran en la siguiente figura.

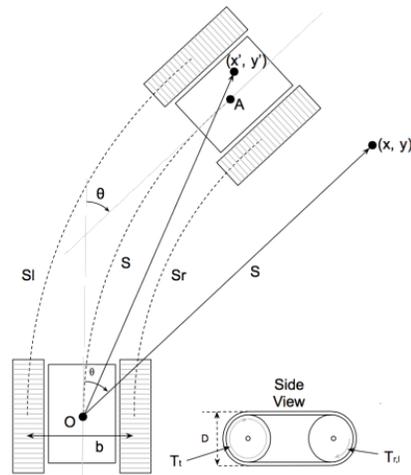


Figura 131 Parámetros de odometría de un robot diferencial

Fuente: (Solderspot, 2017)

Para calcular los parámetros de la odometría, son necesarias las siguientes ecuaciones para describir el modelo cinemático del robot diferencial de acuerdo a (Solderspot, 2017):

$$S_D = \frac{D * \pi * T_D}{T_t} \quad (35)$$

$$S_I = \frac{D * \pi * T_I}{T_t} \quad (36)$$

$$S = \frac{S_D + S_I}{2} \quad (37)$$

$$\theta = \frac{S_I - S_D}{b} \quad (38)$$

$$x = S * \cos(\theta) \quad (39)$$

$$y = S * \sen(\theta) \quad (40)$$

Dónde:

D: Diámetro de la rueda

b: Distancia entre las ruedas

T_t : Resolución del sensor (número de pulsos correspondientes a una vuelta completa)

T_D, T_I : Número de pulsos correspondiente a la rueda derecha e izquierda respectivamente.

S_D, S_I, S : Parámetros de la odometría correspondientes a desplazamientos lineales de la rueda derecha, izquierda y total respectivamente.

x, y, θ : Resultado de la odometría, coordenadas y orientación del robot diferencial.

Ventajosamente, (Solderspot, 2017) publica su librería para entornos de C/C++ e incluido Arduino, la ventaja de esta librería es que es eficiente con la memoria e incluye ciclos de muestreo constantes, además presentan una interfaz amigable para datos de las interrupciones de los pines de Arduino (Donde se encuentran los encoders) de forma nativa. Para configurar la librería, es necesario añadir en el Sketchbook del proyecto, el archivo Navigator.h y Navigator.cpp para incluir los datos y funciones necesarias para generar la localización, el proceso de configuración de la librería se muestra a continuación.

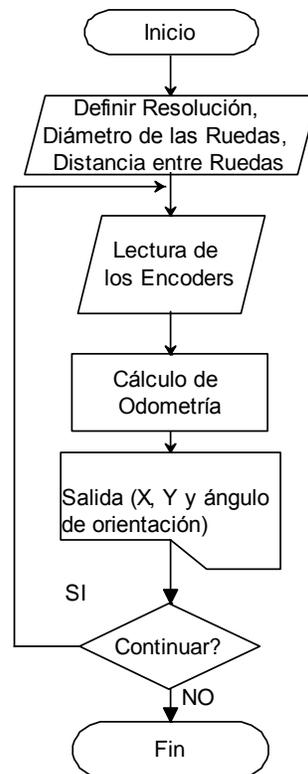


Figura 132 Algoritmo para cálculo de Odometría

b. Control de trayectoria

Una vez con la lectura de las posiciones del prototipo, éste debe tener la capacidad de seguir una trayectoria definida, para lo cual se debe controlar su dirección hacia el punto deseado. Según (RidgeSoft, 2005), se puede utilizar relaciones trigonométricas para determinar en cada paso la dirección del prototipo hacia el próximo destino. Para esto, es necesario encontrar el error en X, el error en Y.

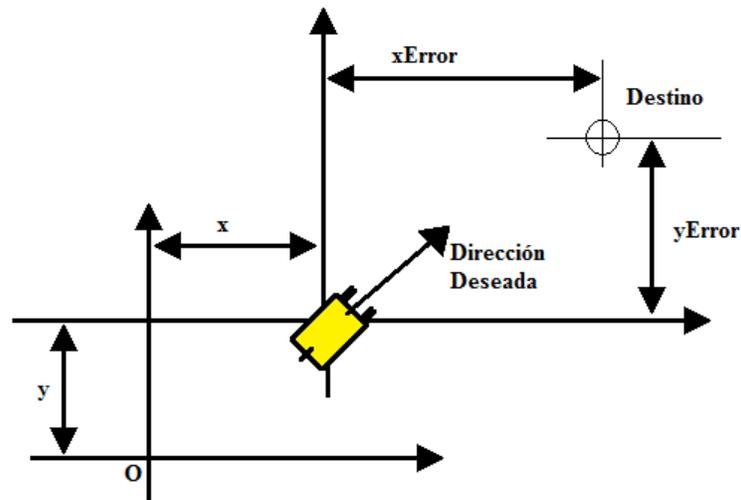


Figura 133 Visualización de los errores en X y en Y

Para lo cual se aplican las siguientes ecuaciones:

$$xError = xDestino - xActual \quad (41)$$

$$yError = yDestino - yActual \quad (42)$$

$$dirección_deseada = ArcTan\left(\frac{yError}{xError}\right) \quad (43)$$

Siendo esta la variable a controlar para que el prototipo consiga llegar al punto deseado, hay que considerar que cuando al prototipo se le aplica el mismo voltaje a los dos motores de las ruedas, éste tendrá una trayectoria recta y si se aplica un mayor voltaje a cada uno de los motores, la plataforma móvil navegará hacia la derecha o izquierda.

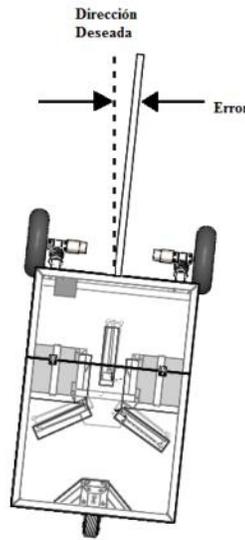


Figura 134 Visualización del error en la dirección del prototipo

Luego de esto es necesario utilizar un tipo de control, para esto se opta por un controlador de tipo proporcional, debido a su amplio uso en sistemas dinámicos, rápida acción y su fácil implementación. Este control posee un diagrama de bloques como se muestra en la siguiente figura:

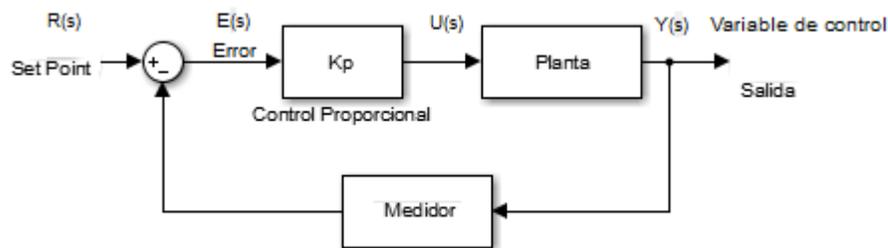


Figura 135 Diagrama de bloques de un control proporcional

Conocido el diagrama de bloques se debe pasarlo a un diagrama de flujo para su posterior implementación en el programa de Arduino. En el diagrama de flujo hay que considerar tanto la limitación del valor máximo y mínimo de la salida, así como su utilización para controlar la velocidad de cada rueda, este diagrama se muestra en la figura 136:

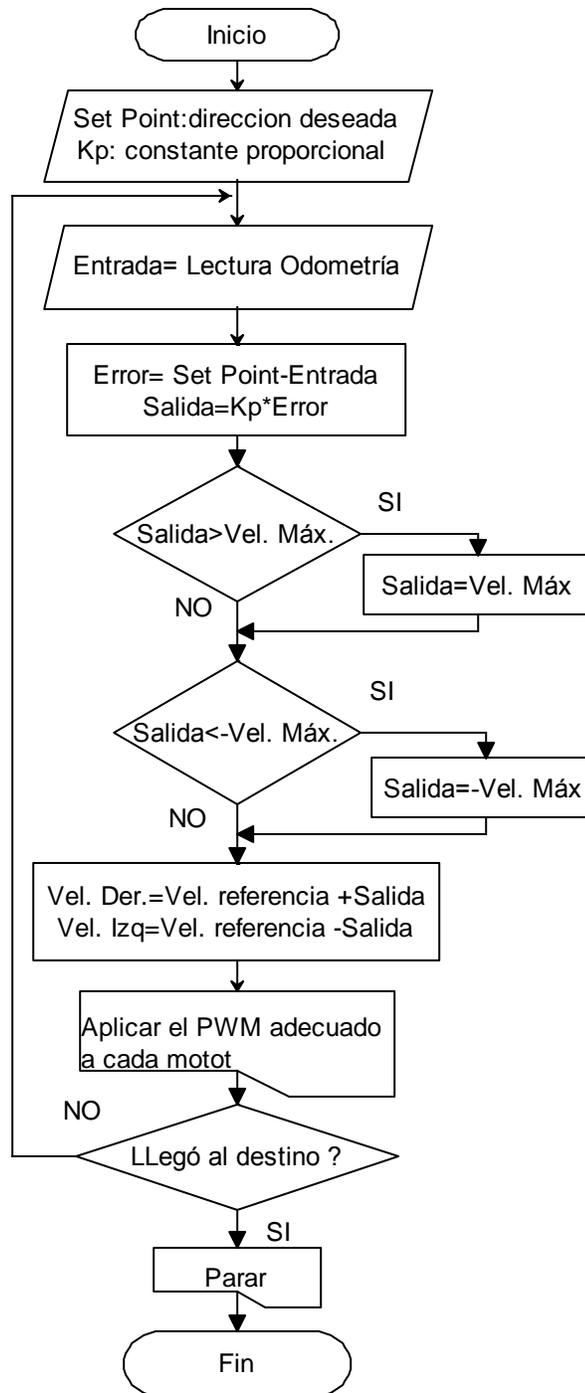


Figura 136 Diagrama de flujo del control proporcional para la navegación

Para encontrar el valor de la constante proporcional se utilizó Matlab® para visualizar las gráficas del set point(color verde)y la entrada(azul).

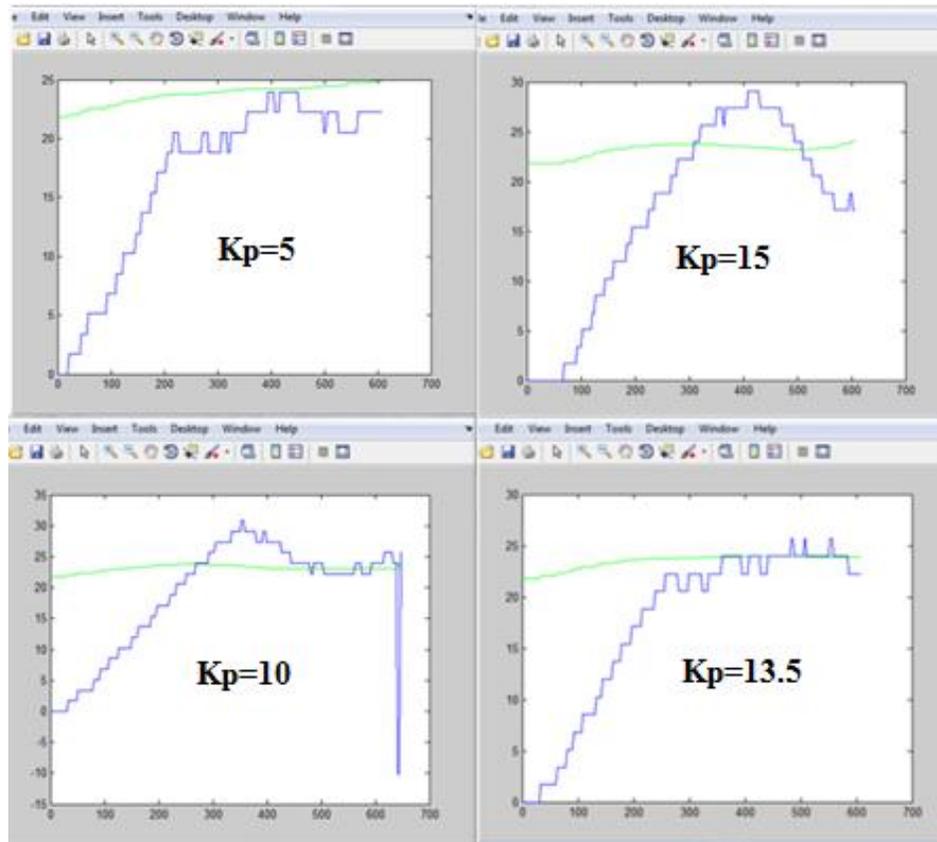


Figura 137 Gráficas del control proporcional con distintas constantes K_p

Una vez obtenidas las gráficas se observa que el valor de la constante proporcional más óptimo es el de 13.5, por lo que se elige este valor para realizar el control.

4.4 Diseño del controlador para la eliminación de mala hierba

4.4.1 Selección de datos para el robot Delta

a. Ciclo de trabajo del robot Delta

Una vez obtenidas las hileras de cultivo, la parte faltante, es obtener los datos que se deben enviar al robot Delta para que realice la acción de deshierba. Se debe considerar que el robot Delta no debe trabajar cerca de las plantas, ni a los lados exteriores de la hilera detectada, por lo que se seleccionan los datos a transmitir al Arduino. Como se vio en la sección 4.1.2 el campo de visión del robot

en la coordenada y es de 1143mm, este valor sobrepasa el campo de trabajo del robot Delta, por lo que se considera separar el área de captura en tres cuadrantes a lo largo de la imagen, con esta división el robot Delta tendrá que trabajar a 381mm en cada área de trabajo temporal, logrando un menor esfuerzo para los componentes del Delta y así ahorrar energía. El diagrama del área de trabajo es el que se nota a continuación:

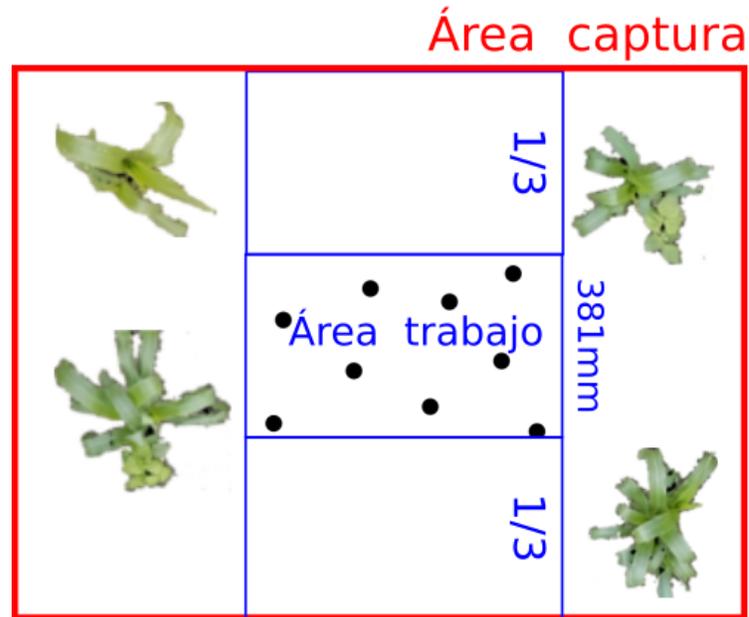


Figura 138 Área trabajo del robot Delta

Para lograr que el robot Delta se ubique sobre el área de trabajo temporal, es necesario considerar que hay una distancia entre el centro de la cámara y el centro del robot Delta, entonces para trabajar un área temporal se debe recurrir a realizar cuatro ciclos como muestra la siguiente figura:

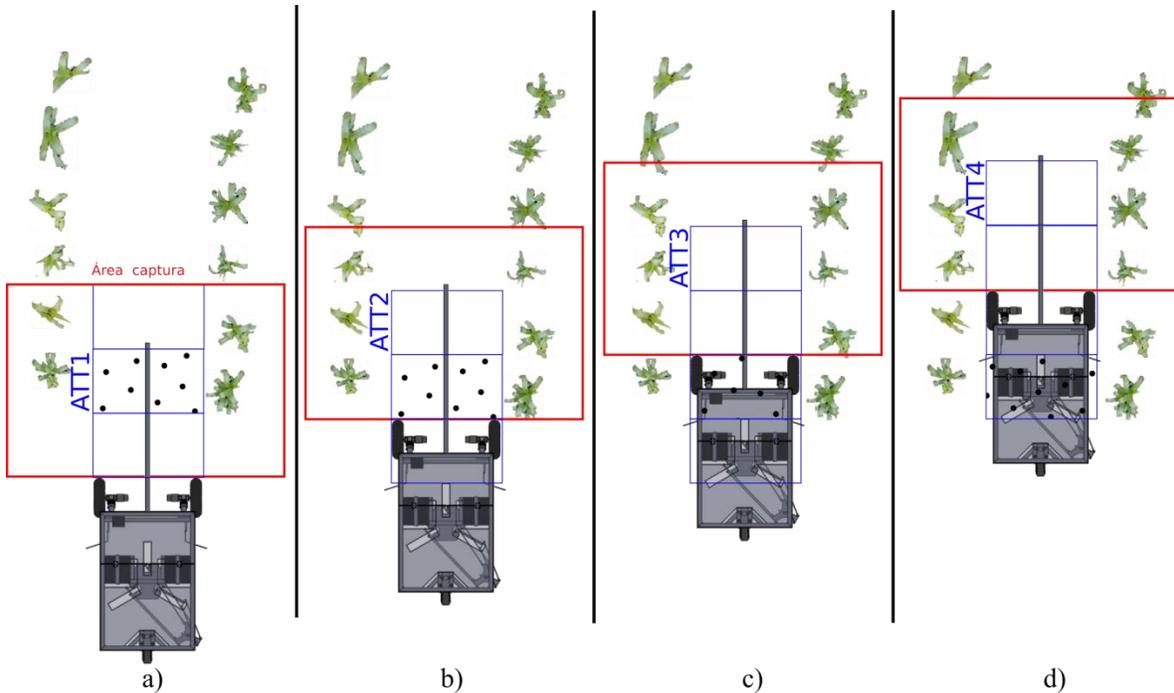


Figura 139 Ciclo de procesamiento para la actuación del robot Delta

Cómo se nota, en *a)* el robot Delta procesa la imagen del área de trabajo temporal (ATT) 1, una vez que obtiene los datos de los elementos de la mala hierba, los envía para que se almacene en el Arduino, luego la plataforma móvil hace su primer desplazamiento. En *b)* el robot ha hecho su primer desplazamiento, la cámara toma otra imagen y procesa el ATT 2, el centro de robot Delta se encuentra a dos ciclos de posicionarse sobre el área temporal 1, envía los datos e inicia otra navegación. En *c)* la cámara toma otra imagen y procesa el ATT 3, el centro de robot Delta se encuentra todavía a un ciclo del área temporal 1, se envía los datos y se navega. En *d)* el centro del robot Delta se posiciona sobre el área de trabajo temporal 1, empieza a trabajar sobre ella y al mismo tiempo se toma la imagen del área de trabajo temporal 4 y se procesa, una vez finalizada el trabajo del Delta, se envía los datos del ATT4 y se navega, el ciclo es repetitivo y en el siguiente ciclo se trabajará sobre el ATT2, entonces se puede concluir que después de tres ciclos se alcanza el área procesada para que trabaje el robot Delta.

b. Eliminación de datos innecesarios

Cómo se señaló en el anterior punto, los datos que se envían al Delta, solo se enviará la tercera parte central de la imagen que se encuentre dentro de la hilera, por lo que es necesario eliminar los contornos que queden fuera de la misma, entonces, un ejemplo de imagen sería como en la siguiente figura:

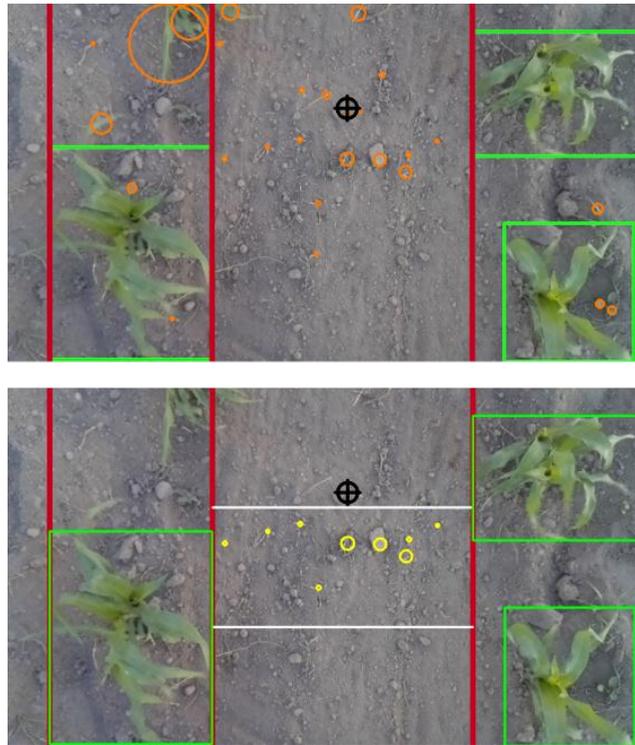


Figura 140 Arriba: Datos totales Abajo: Datos necesarios

Cómo se nota en la anterior figura, el primer procesamiento deja una cantidad de datos muy grande, aparte algunos datos se encuentran dentro del maíz, por lo que son descartados, la nueva distribución de datos que se presenta abajo es más compacta y cumple con el área de trabajo del robot Delta, por lo que sólo es necesario recurrir a su transmisión.

4.4.2 Transmisión de datos

Con los datos necesarios, se procede a transmitirlos hacia el Arduino, como en la anterior transmisión, primero se envía el número de puntos de mala hierba detectada en un vector de 2 caracteres para que el Arduino almacene en un vector del mismo tamaño, luego en un dato de 6 cifras (3 para la coordenada x del centro y 3 para la coordenada y del centro), se envían todas las coordenadas transformadas a mm. Cuando todas las coordenadas se envían al Arduino, éste responde con un carácter de confirmación al Raspberry Pi para que pueda seguir con el siguiente paso.

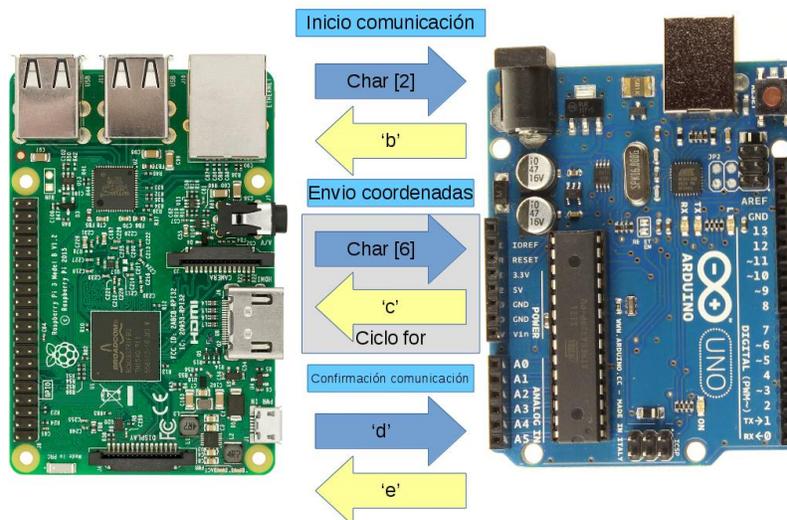


Figura 141 Diagrama de bloques de la comunicación

4.4.3 Diseño del Control del Robot Delta

Para la realización del control del robot tipo Delta, se tiene que desarrollar su cinemática inversa, para ello es preciso conocer el tipo de configuración del robot paralelo, de esta forma se obtendrán las ecuaciones que rijan su movimiento, la cual, según (Salazar & Pachacama, 2006) se la puede desarrollar de la siguiente manera:

Primero se establece un sistema de referencia global, representado por $OXYZ$, que será ubicado en el centro de la base fija del robot, el punto $P(P_x, P_y, P_z)$ está ubicado en el centro de la parte móvil. Los eslabones están representados con L_1 y L_2 respectivamente. Como se muestra en la siguiente imagen:

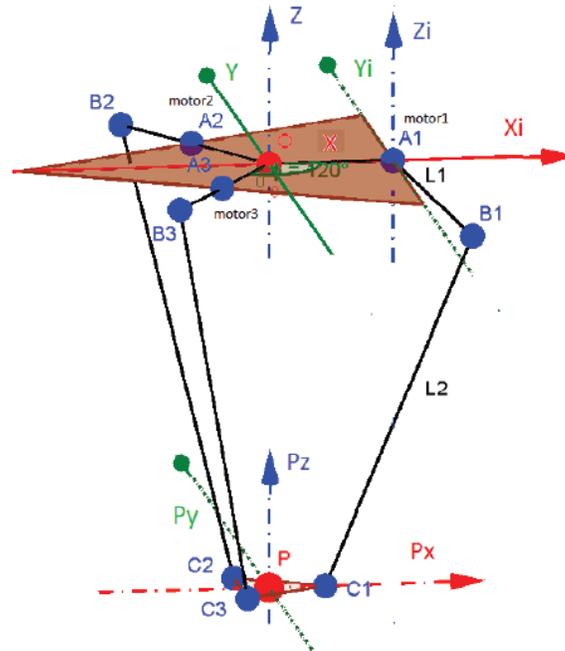


Figura 142 Modelo de la geometría de un Robot Delta

Fuente: (Salazar & Pachacama, 2006)

Sobre cada cadena cinemática se selecciona un sistema de coordenadas referencia $O'X_iY_iZ_i$, el cual está localizado a una distancia R del sistema de referencia global y rotada un ángulo $\phi_i (i = 1,2,3)$ correspondiente a $0^\circ, 120^\circ, 240^\circ$ respectivamente. Se utiliza una matriz de transformación T , para poder representar la orientación y posición de $O'X'Y'Z'$, el cual estará rotado y trasladado respecto al sistema de referencia global. Esta matriz permitirá encontrar las coordenadas del vector $\vec{r} = [r_x, r_y, r_z]^T$ en el sistema global, partiendo del vector $\vec{r}' = [r'_x, r'_y, r'_z]^T$:

$$\begin{bmatrix} r_x \\ r_y \\ r_z \end{bmatrix} = T \begin{bmatrix} r'_x \\ r'_y \\ r'_z \end{bmatrix} \quad (44)$$

El sistema $A_i X_i Y_i Z_i$, se encuentra trasladado una distancia R a lo largo del eje OX y rotado ϕ_i alrededor del eje OZ con respecto al eje de referencia global $OXYZ$. Como se muestra en la siguiente imagen:

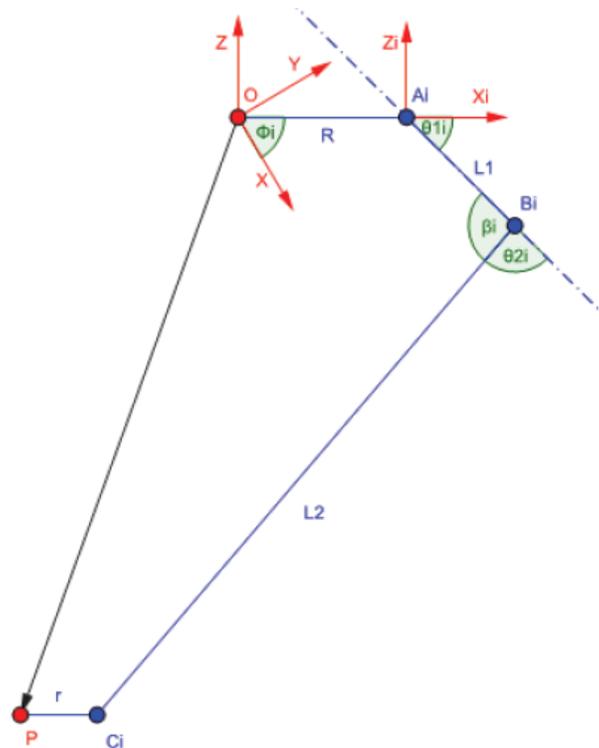


Figura 143 Ángulos de una cadena cinemática del robot Delta

Fuente: (Salazar & Pachacama, 2006)

Esto nos permite obtener los vectores $\overrightarrow{OA_i}$, $\overrightarrow{OB_i}$ y $\overrightarrow{OC_i}$ a partir de los vectores $\overrightarrow{A_i B_i}$ y de $\overrightarrow{A_i C_i}$. Para lo cual es necesario una matriz de transformación correspondiente a una traslación seguida de una rotación sobre el eje Z , la cual está definida por:

$$T(P, (Z, \alpha)) = \begin{bmatrix} \cos(\alpha) & -\text{sen}(\alpha) & 0 & P_x \cos(\alpha) - P_y \text{sen}(\alpha) \\ \text{sen}(\alpha) & \cos(\alpha) & 0 & P_x \text{sen}(\alpha) + P_y \cos(\alpha) \\ 0 & 0 & 1 & P_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (45)$$

Aplicando esta matriz, donde $P_x = R$, $P_y = 0$, $P_z = 0$ y el ángulo de rotación $\alpha = \phi_i$ se obtiene:

$$\overrightarrow{OA_i} = \begin{bmatrix} \cos \phi_i & -\text{sen} \phi_i & 0 & R \cos \phi_i \\ \text{sen} \phi_i & \cos \phi_i & 0 & R \text{sen} \phi_i \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} R \cos(\phi_i) \\ R \text{sen}(\phi_i) \\ 0 \\ 1 \end{bmatrix} \quad (46)$$

$$\overrightarrow{OB_i} = \begin{bmatrix} \cos \phi_i & -\text{sen} \phi_i & 0 & R \cos \phi_i \\ \text{sen} \phi_i & \cos \phi_i & 0 & R \text{sen} \phi_i \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} L_1 \cos \theta_{1i} \\ 0 \\ -L_1 \text{sen} \theta_{1i} \\ 1 \end{bmatrix} = \begin{bmatrix} \cos \phi_i (L_1 \cos \theta_{1i} + R) \\ \text{sen} \phi_i (L_1 \cos \theta_{1i} + R) \\ -L_1 \text{sen} \theta_{1i} \\ 1 \end{bmatrix} \quad (47)$$

$$\overrightarrow{OC_i} = \begin{bmatrix} \cos \phi_i & -\text{sen} \phi_i & 0 & R \cos \phi_i \\ \text{sen} \phi_i & \cos \phi_i & 0 & R \text{sen} \phi_i \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} P_x \cos \phi_i + P_y \text{sen} \phi_i + r - R \\ P_y \cos \phi_i + P_x \text{sen} \phi_i \\ P_z \\ 1 \end{bmatrix} = \begin{bmatrix} P_x + r \cos \phi_i \\ P_y + r \text{sen} \phi_i \\ P_z \\ 1 \end{bmatrix} \quad (48)$$

Estos vectores permiten encontrar los vectores $\overrightarrow{B_i A_i}$, $\overrightarrow{B_i C_i}$ y $\overrightarrow{A_i C_i}$

$$\overrightarrow{B_i A_i} = \overrightarrow{OA_i} - \overrightarrow{OC_i} = \begin{bmatrix} \cos \phi_i L_1 \cos \theta_{1i} \\ -\text{sen} \phi_i L_1 \cos \theta_{1i} \\ L_1 \text{sen} \theta_{1i} \end{bmatrix} \quad (49)$$

$$\overrightarrow{B_i C_i} = \overrightarrow{OC_i} - \overrightarrow{OB_i} = \begin{bmatrix} P_x - \cos \phi_i (R + L_1 \cos \theta_{1i} - r) \\ P_y - \text{sen} \phi_i (R + L_1 \cos \theta_{1i} - r) \\ P_z + L_1 \text{sen} \theta_{1i} \end{bmatrix} \quad (50)$$

$$\overrightarrow{A_i C_i} = \overrightarrow{OC_i} - \overrightarrow{OA_i} = \begin{bmatrix} P_x - \cos \phi_i (r - R) \\ P_y - \text{sen} \phi_i (r - R) \\ P_z \end{bmatrix} \quad (51)$$

Una vez obtenidos estos vectores se procede a realizar un análisis geométrico y vectorial de una cadena cinemática:

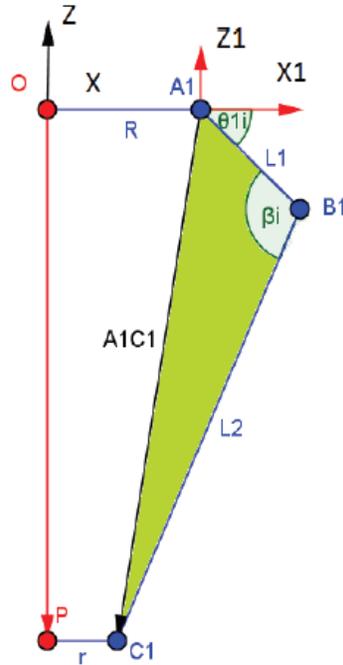


Figura 144 Representación vectorial de la primera cadena cinemática

Fuente: (Salazar & Pachacama, 2006)

De la imagen anterior se observa los vértices del triángulo formado $A_i B_i C_i$, de donde aplicando ley de cosenos, se puede hallar la longitud de $A_i C_i$.

$$A_i C_i^2 = L_1^2 + L_2^2 - 2L_1 L_2 \cos(\beta_i) \quad (52)$$

Para encontrar el valor del $\cos(\beta_i)$ se aplica la definición de producto escalar:

$$\overline{B_i A_i} * \overline{B_i C_i} = \|B_i A_i\| \|B_i C_i\| \cos(\beta_i) \quad (53)$$

Como $L_1 = \|B_i A_i\|$ y $L_2 = \|B_i C_i\|$, se llega a:

$$\cos(\beta_i) = \frac{\overline{B_i A_i} * \overline{B_i C_i}}{L_1 L_2} \quad (54)$$

Reemplazando (54) en (52) se obtiene:

$$A_i C_i^2 = L_1^2 + L_2^2 - 2(\overline{B_i A_i} * \overline{B_i C_i}) \quad (55)$$

Multiplicando (49) y (50) se obtiene:

$$\overrightarrow{B_i A_i} * \overrightarrow{B_i C_i} = L_1^2 - L_1 \cos \theta_{1i} (P_x \cos \phi_i + P_y \sin \phi_i + r - R) - L_1 \sin \theta_{1i} P_z \quad (56)$$

Reemplazando (56) en (55) se obtiene:

$$A_i C_i^2 = L_1^2 - L_2^2 + 2L_1 \cos \theta_{1i} (P_x \cos \phi_i + P_y \sin \phi_i + r - R) - 2L_1 \sin \theta_{1i} P_z \quad (57)$$

La magnitud del vector $\overrightarrow{A_i C_i}$ de la ecuación (51) es:

$$\|\overrightarrow{A_i C_i}\| = A_i C_i^2 = (P_x + \cos \phi_i (r - R))^2 + (P_y + \sin \phi_i (r - R))^2 + P_z^2 \quad (58)$$

Igualando (58) y (57) se obtiene la ecuación de la forma:

$$a_i \sin \theta_{1i} + b_i \cos \theta_{1i} = c_i \quad (59)$$

Dónde:

$$a_i = -2P_z L_i$$

$$b_i = 2L_i (P_x \cos \phi_i + P_y \sin \phi_i + r - R)$$

$$c_i = (P_x + \cos \phi_i (r - R))^2 + (P_y + \sin \phi_i (r - R))^2 + P_z^2 + L_1^2 - L_2^2$$

Tomando en cuenta las identidades trigonométricas de la suma de senos y cosenos se obtiene:

$$a_i \sin \theta_{1i} + b_i \cos \theta_{1i} = R \sin(\theta_{1i} - \alpha) \quad (60)$$

Dónde:

$$R = \sqrt{a_i^2 + b_i^2}$$

$$\alpha = \tan^{-1} \left(\frac{b_i}{a_i} \right)$$

Reemplazando la ecuación (60) en (59) se obtiene:

$$\sqrt{a_i^2 + b_i^2} \sin \left(\theta_{1i} + \tan^{-1} \left(\frac{b_i}{a_i} \right) \right) = c_i \quad (61)$$

Resolviendo la ecuación (61) se obtiene la solución

$$\theta_{1i} = \sin^{-1} \left(\frac{c_i}{\sqrt{a_i^2 + b_i^2}} \right) - \tan^{-1} \left(\frac{b_i}{a_i} \right) \quad (62)$$

Una vez obtenida, la fórmula para encontrar la solución de la cinemática inversa, se procede a realizar el diagrama de flujo correspondiente al control del robot Delta.

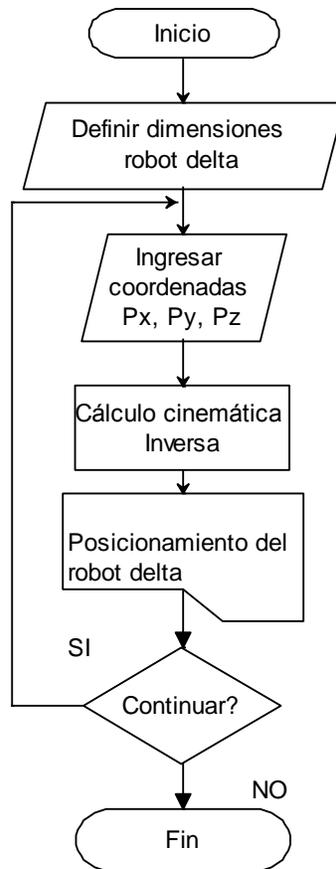


Figura 145 Diagrama de flujo para el control del robot Delta

Entonces, una vez que se ha definido todas las secuencias de control, se presenta el diagrama final de control, que integra la función del Raspberry Pi y del Arduino (ver figura 146)

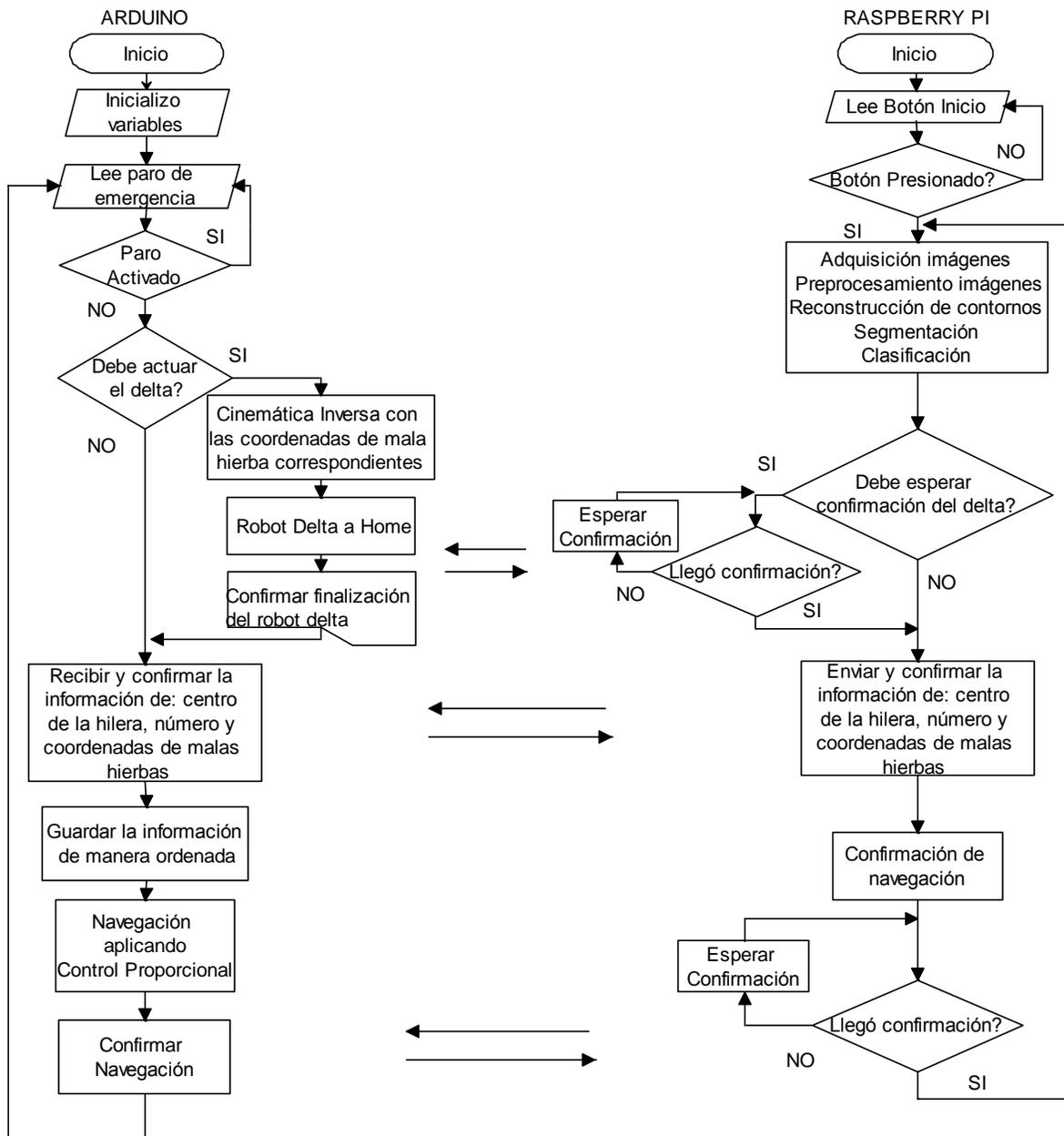


Figura 146 Diagrama de control total del prototipo

Finalmente el programa del Raspberry Pi se hace ejecutable al inicio, a través de un demonio (palabra que viene de la nomenclatura UNIX para programas residentes o servicios) por el Systemd.

CAPÍTULO V

PRUEBAS Y RESULTADOS

5.1 Descripción del campo de prueba

5.1.1 Localización

Para realizar las pruebas del prototipo de robot de deshierba, se consideró al terreno proporcionado por la Asociación de Agricultores “Unión y Progreso”. El campo está ubicado en el Barrio Yacupamba, Parroquia San Miguelito, del Cantón Santiago de Píllaro – Tungurahua – Ecuador. El terreno posee la condición de no ser pedregoso además que es plano con poca pendiente, que es la condición principal para el trabajo del robot.

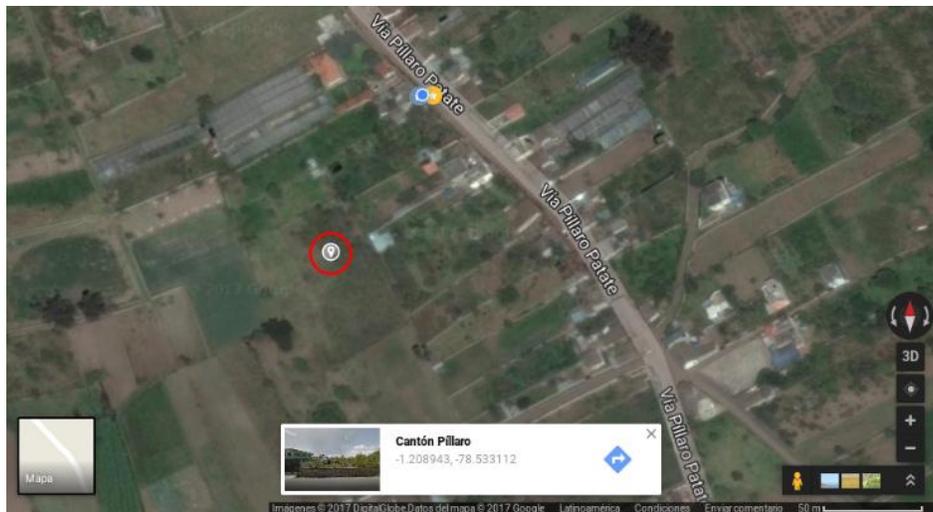


Figura 147 Ubicación del terreno de maíz

5.1.2 Cultivo de maíz

En cuanto a las plantas de maíz, el terreno cuenta con dos plantaciones de maíz de manera escalonada (sembrados con 15 días de diferencia), que permitirá demostrar la efectividad de deshierba del prototipo en estas dos condiciones que son las otorgadas por el SINAGAP, que la deshierba debe ser a los 30 y a los 45

días después de la plantación. En la siguiente figura se muestra el estado de la plantación al día de las pruebas.



Figura 148 Izquierda: Maíz V7, Derecha: Maíz V10

El estado de mala hierba en el terreno es totalmente aleatorio y corresponde a múltiples especies de plantas, su desarrollo es variable e igual depende de cada especie de planta, además algunos muestran raíces más o menos profundas, la distribución de la mala hierba, es cómo se muestra en la siguiente figura:



Figura 149 Plantas del terreno de cultivo

5.2 Pruebas del funcionamiento prototipo

5.2.1 Detección y clasificación

Las pruebas que se presentan a continuación reflejan los resultados de procesamiento de visión artificial y clasificación, se tomó 25 imágenes del campo de maíz, por lo que a continuación se presenta los resultados de los distintos las distintas pruebas.

a. Pruebas de rendimiento

Esta prueba es la encargada de demostrar la eficiencia y rapidez del microcomputador, se pasa por el sistema todas las imágenes y los resultados del tiempo para cada proceso son los que se muestran en la siguiente tabla.

Tabla 26
Tiempo de procesamiento por etapa

Tiempo de procesamiento en segundos						
Nº	Pre-procesamiento	Reconstrucción de contornos	Clasificación	Generación hilera	Eliminar innecesarios	Total
1	0,27	0,18	2,75	0,02	0,27	3,48
2	0,27	0,12	3,93	0,01	0,27	4,59
3	0,25	0,34	6,39	0,02	0,25	7,25
4	0,25	0,14	4,02	0,00	0,25	4,67
5	0,25	0,19	6,28	0,00	0,25	6,97
6	0,27	1,50	4,40	0,01	0,27	6,44
7	0,25	0,19	6,32	0,00	0,25	7,02
8	0,25	0,20	6,42	0,01	0,25	7,12
9	0,25	0,14	4,12	0,01	0,25	4,77
10	0,27	0,18	2,74	0,01	0,27	3,46
11	0,27	0,12	4,01	0,01	0,27	4,68
12	0,25	0,34	6,34	0,01	0,25	7,19
13	0,25	0,19	6,38	0,01	0,25	7,08

CONTINÚA



15	0,25	0,04	2,60	0,01	0,25	3,14
16	0,25	0,19	6,63	0,01	0,25	7,33
17	0,25	0,19	6,24	0,02	0,25	6,95
18	0,25	0,06	2,73	0,02	0,25	3,32
19	0,25	0,07	5,39	0,01	0,25	5,96
20	0,25	0,05	2,41	0,01	0,25	2,97
21	0,25	0,05	5,45	0,00	0,25	6,00
22	0,25	0,17	4,03	0,00	0,25	4,70
23	0,25	0,06	2,46	0,00	0,25	3,02
24	0,25	0,10	3,94	0,01	0,25	4,55
25	0,26	0,05	2,48	0,01	0,26	3,05
Tiempo promedio de procesamiento						5,22



Figura 150 Distribución de los tiempos de procesamiento

Como se puede apreciar en la anterior etapa, el proceso que toma más tiempo es el de clasificación (un 86% del tiempo total), pues tiene que hacer un cómputo de todos los parámetros de la red. Dependiendo del estado de la imagen puede variar el tiempo de procesamiento para las otras tres etapas (pre-procesamiento, eliminar innecesarios y reconstrucción de contornos), el tiempo promedio es de menos de 6 segundos que parece mucho, pero cabe destacar que

éste se realiza mientras el robot Delta está realizando su trabajo, por lo que su impacto es mínimo.

b. Pruebas de eficiencia

Cómo la mayoría de procedimientos de visión, se basa en el análisis de contornos de una imagen binaria, por lo que se analiza el número de contornos a lo largo de cada etapa y su reducción final, un ejemplo del procesamiento se puede observar en la siguiente imagen:

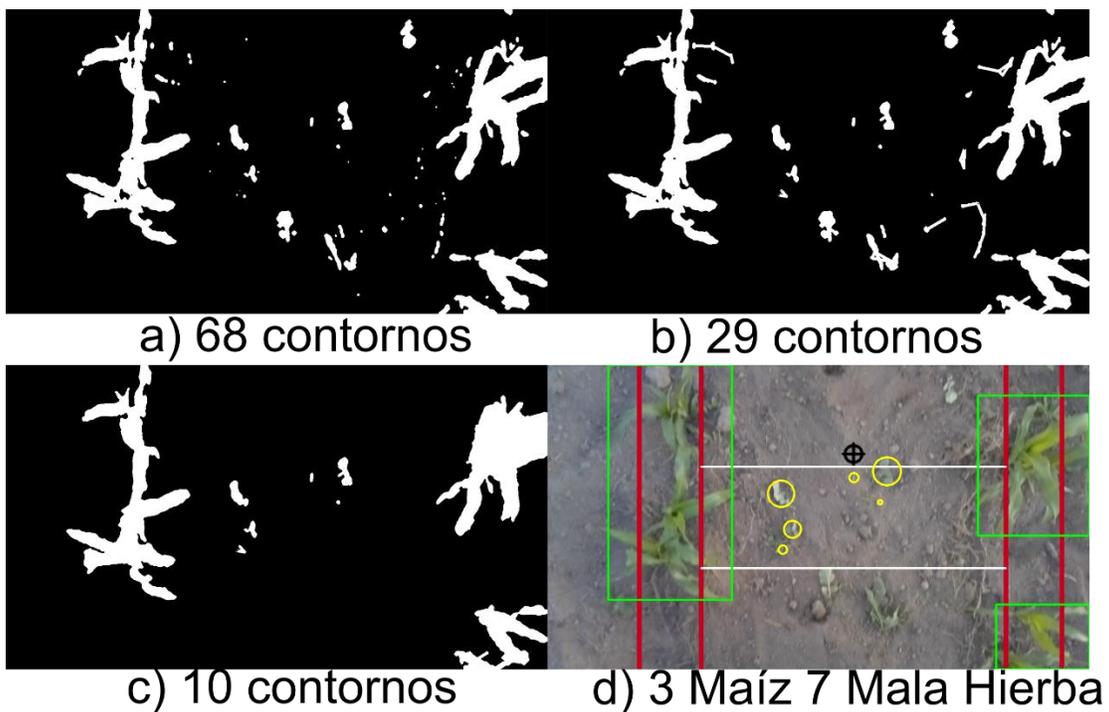


Figura 151 Proceso de reducción de contornos

Con las mismas 25 imágenes de prueba, se realiza el proceso de reducción de contornos, se distinguen cuatro etapas como se observa en la imagen anterior, donde a) es el pre-procesamiento de la imagen, b) la reducción de contornos, c) la eliminación de contornos innecesarios para enviar los datos al robot Delta y d) la clasificación final, el resultado se puede observar de mejor forma en la tabla 27.

Tabla 27
Reducción de contornos por cada etapa

Reducción de contornos por etapa en número de contornos					
Nº	Pre-procesamiento	Reconstrucción de contornos	Eliminar innecesarios	Final	
				Maíz	Mala hierba
1	20	12	1	1	0
2	90	43	13	3	10
3	23	16	4	3	1
4	68	29	10	3	7
5	34	24	5	3	2
6	41	25	2	2	2
7	34	24	5	3	2
8	34	24	5	3	2
9	68	29	10	3	7
10	20	12	1	1	0
11	90	43	13	3	10
12	23	16	4	3	1
13	34	24	5	3	2
15	24	11	1	1	0
16	34	24	5	3	2
17	34	24	5	3	2
18	24	12	4	2	2
19	20	13	4	4	2
20	20	13	3	2	1
21	19	13	5	3	2
22	27	8	4	3	1
23	25	17	3	2	1
24	25	16	3	3	0
25	20	11	3	2	1

La tabla anterior refleja que el proceso de reconstrucción y eliminación de contornos innecesarios es efectivo, se logra hasta una media del 86% de contornos reducidos, con lo que se mejora la detección de las plantas y además al eliminar los puntos innecesarios se puede dar menos trabajo al robot Delta, con lo que se ahorraría tiempo y energía, haciendo que el proceso de deshierba de grandes cultivos sea eficaz. En la siguiente figura se ve la distribución de los contornos reducidos del total:

Porcentaje del reducción contornos



Figura 152 Porcentaje de reducción de contornos

5.2.2 Navegación

El valor final de la prueba de navegación es muy importante, debido a que este proceso permite que el robot no choque contra las plantas, para comprobar que haya una aceptable navegación, es necesario comprobar que el valor medido sea cercano al valor calculado por el proceso de odometría, los resultados se presentan en la siguiente tabla:

Tabla 28
Pruebas de navegación en superficie lisa

Pruebas de navegación(mm)				
Prueba	X medido	Y medido	X calculado	Y calculado
1	7	375	5,03	371,77

CONTINÚA 

2	5	370	0,42	364,77
3	-4	368	6,07	371,73
4	-20	370	9,63	371,64
5	-3	370	1,26	374,77
6	-3	370,6	4,482	370,93
7	2	369	4,432	372,8
8	4	371	2,56	369,1
9	-2	372	3,29	374,55
10	1	368	7,57	370,93
Media	1,30	370,36	4,47	371,39

Cómo se observa en la tabla anterior, los valores de navegación son muy cercanos a los valores calculados, con una diferencia media de 3.17 mm en el eje x y una diferencia de 1,03 mm en el eje y, además como el objetivo del robot es el terreno agrícola, entonces se presenta pruebas en el terreno agrícola.



Figura 153 Pruebas de navegación en el terreno de cultivo

Tabla 29
Pruebas de navegación en el terreno de cultivo (en mm)

Pruebas de navegación(mm)				
Prueba	X medido	Y medido	X calculado	Y calculado
1	10	350	0,84	371,77
2	30	380	0	385
3	30	320	0,48	376
4	20	375	1,47	378
5	25	360	-0,42	378
6	10	380	0,21	378
7	25	375	1,05	378
8	-10	363	-0,21	378
9	-10	375	0,21	378
10	-20	365	0	378
Media	11	364,3	0,342	340,7

Los valores obtenidos son muy acertados en la realidad, debido a la configuración del robot el error es inevitable pero aceptable para el prototipo, se observa una diferencia media de 11mm en la coordenada x y una diferencia de 20mm en la coordenada y, que es aceptable entonces se aceptan estos resultados como óptimos para la plataforma móvil.

5.2.3 Robot Delta

a. Comparación entre valores teóricos y medidos

Para calibrar el robot Delta se procede a tomar mediciones de los ángulos de los servomotores al variar la coordenada Z y en la posición (0,0) en el plano XY

y se los compara con los valores teóricos, esto se puede observar en la siguiente tabla:

Tabla 30
Pruebas de posicionamiento del robot Delta

Mediciones de los ángulos del robot Delta en grados(°)							
mm		Servo1(°)		Servo2(°)		Servo3(°)	
Zmedido	Zteórico	Medido	Teórico	Medido	Teórico	Medido	Teórico
332	-335	-62	-61,75	-63,5	-61,75	-67,8	-61,75
341	-340	-58	-56,74	-60,5	-56,74	-63,5	-56,74
345	-345	-53	-52,47	-55	-52,47	-60,9	-52,47
349	-350	-49	-48,7	-50	-48,7	-54,9	-48,7
352	-355	-45	-45,27	-46,2	-45,27	-51,5	-45,27
359	-360	-42,5	-42	-43	-42	-48,8	-42
364	-365	-40	-39,19	-40,5	-39,19	-45	-39,19
370	-370	-36	-36,43	-37	-36,43	-41,3	-36,43
376	-375	-33	-33,82	-33,8	-33,82	-37,5	-33,82
382	-380	-31,5	-31,33	-31,1	-31,33	-35,1	-31,33

Como se puede apreciar existen diferencias entre los valores teóricos con los medidos, esto puede deberse a fallas en la construcción mecánica, o por el

juego de los acoples del servomotor. Para solucionar el error de posicionamiento se debe considerar el promedio de las diferencias de ángulos teóricos y medidos de cada servomotor, los cuales se muestran a continuación:

Tabla 31
Diferencias entre los ángulos de los servomotores

Diferencia entre ángulos teóricos y medidos(°)			
Nº	Servo1	Servo2	Servo3
1	0,25	1,75	6,05
2	1,26	3,76	6,76
3	0,53	2,53	8,43
4	0,3	1,3	6,2
5	-0,27	0,93	6,23
6	0,5	1	6,8
7	0,81	1,31	5,81
8	-0,43	0,57	4,87
9	-0,82	-0,02	3,68
10	0,17	-0,23	3,77
Promedio	0,23	1,29	5,86

Una vez encontrado los promedios, se usa estos valores y se corrige el posicionamiento de los servos mediante programación, de esta manera se logra una mayor precisión en el robot tipo Delta.

b. Comprobación del espacio de trabajo

Como se mencionó en el Capítulo 3 el robot tipo Delta teóricamente posee un espacio de trabajo limitado de -300 a 300 mm en los ejes X e Y, cuando Z esté dentro del rango de -340 a -440 mm, establecido en este proyecto como la zona

de seguridad y de trabajo respectivamente, por lo que se procede a comprobarlo posicionando al robot Delta en los puntos extremos, como se aprecia en la siguiente tabla:

Tabla 32
Puntos extremos y ángulos de los servos del robot Delta

Posición efector final (mm)			Posición Servos (grados)		
Px	Py	Pz	Servo1	Servo2	Servo3
-300	-300	-340	74,84	-28,83	59,33
-300	-300	-440	87,31	-8,34	78,70
300	-300	-340	74,84	59,33	-28,83
300	-300	-440	83,71	78,70	-8,34
-300	300	-340	-14,33	20,77	79,36
-300	300	-440	8,29	45,31	89,29
300	300	-340	-14,33	79,36	20,77
300	300	-440	8,29	89,29	45,31

Se puede observar ocho puntos, los que debe ser capaz de alcanzar el efector final del robot tipo Delta, puesto que si alcanza estos, logrará alcanzar cualquier punto dentro del espacio de trabajo. En la siguiente imagen se muestra al robot Delta ubicándose en los puntos extremos.

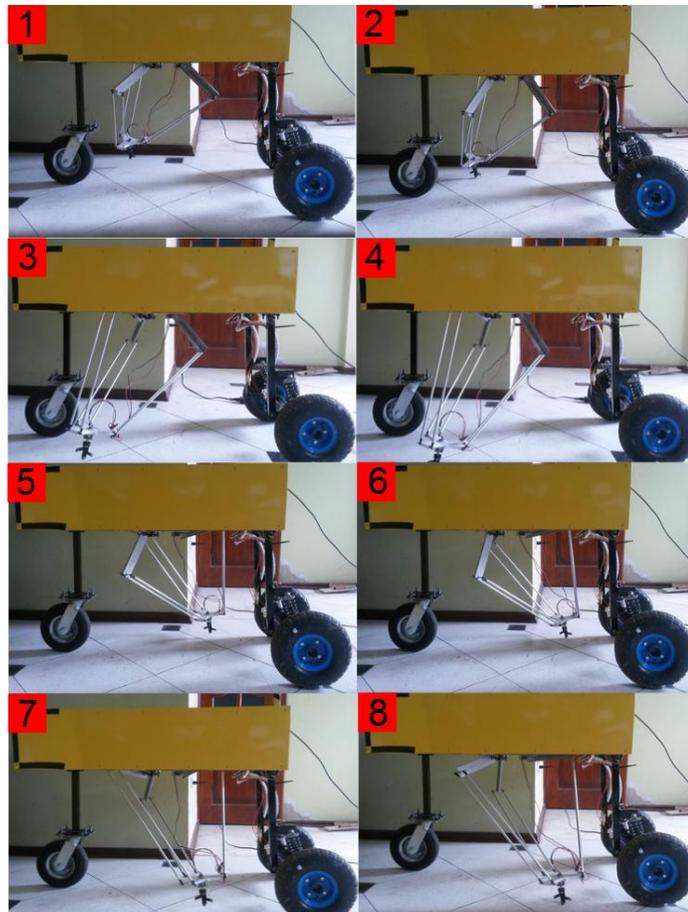


Figura 154 Robot Delta recorriendo el espacio de trabajo

Como se puede observar el robot Delta puede alcanzar los puntos extremos del espacio de trabajo. Cabe destacar que en los puntos más extremos los servomotores consumen más corriente, puesto que el torque requerido es mayor que en otros puntos dentro del área de trabajo.

c. Comprobación de la exactitud de la cinemática inversa

Para comprobar la cinemática del robot tipo Delta es necesario realizar pruebas donde se le ubica al robot paralelo en puntos específicos y se procede a medir el valor real y el teórico, lo cual se ve reflejado en la tabla 33:

Tabla 33
Valores del posicionamiento del robot Delta

Valores teóricos y medidos del posicionamiento del robot Delta (Pz=-340mm)					
Px(mm)			Py (mm)		
Teórico	Medido	Diferencia	Teórico	Medido	Diferencia
-250	-225	25	0	0	0
-200	-173	23	0	-1	1
-150	-130	20	0	-2	2
-100	-85	15	0	-4	4
-50	-40	10	0	-7	7
0	0	0	0	-20	20
50	47	3	0	-12	12
100	95	5	0	-10	10
150	139	11	0	-6	6
200	175	25	0	-4	4
250	210	40	0	0	0
0	24	24	-250	-208	42
0	28	28	-200	-174	26
0	26	26	-150	-142	8
0	15	15	-100	-96	4
0	8	8	-50	-49	1
0	0	0	0	0	0
0	0	0	50	41	9
0	1	1	100	96	4
0	2	2	150	149	1

CONTINÚA 

0	2	2	200	189	11
0	3	3	250	234	16
Promedio en X		16,45	Promedio en Y		11,09

Como se puede apreciar aún se tiene un error tanto eje x de un promedio de 16,45 mm, y de 11,09 mm en el eje y, siendo más crítico en ambos casos en los extremos, pero debido a que la herramienta tiene 55mm de diámetro, se logra compensar el error producido por el posicionamiento.

5.2.4 Deshierba

Para la deshierba, con respecto al origen del robot Delta se ubicó la coordenada en Z a -340mm para la zona de seguridad y a -410mm para la zona de eliminación, lo que corresponde a un promedio -20mm debajo del suelo, lo que afectará la raíz de las plantas no benéficas, por lo que a esta distancia será la adecuada para que trabaje la herramienta de eliminación, el proceso de deshierba se muestra a continuación.

a. Muestra

Para comprobar que tan efectivo es el prototipo, se realiza un muestreo del campo de maíz, se contabilizó 1008 plantas que es la población total (distribuida igualmente entre maíz de etapa vegetativa V7 y V10), se debe recurrir a sacar una muestra aleatoria simple, para ello se calculó el tamaño de la muestra para población finita, que según (Psyma, 2017) es:

$$n = \frac{N * Z_{\alpha}^2 * p * q}{d^2 * (N - 1) + Z_{\alpha}^2 * p * q} \quad (63)$$

Donde:

N: Tamaño de la población

Z_α: El nivel de confianza

p: Probabilidad de éxito

q: Probabilidad de fracaso

d: Precisión (Error máximo admisible en términos de proporción)

Para el presente documento, se realiza el muestreo con un nivel de confianza del 95% ($Z_{\alpha} = 1,96$), una probabilidad de éxito y fracaso igual ($p = q = 0,5$) además de un error admisible del 5%, entonces el cálculo de la muestra es:

$$n = 278$$

Por lo tanto, se debe realizar las pruebas sobre 139 plantas de maíz en la etapa vegetativa V7 y V10.

b. Metodología

Para comparar que tan efectivo fue el proceso de deshierba del robot, se guardan registros de éste en carpetas separadas que contienen la imagen de procesamiento, los datos enviados al robot delta y se cuenta las acciones del robot paralelo, se hacen pruebas de acuerdo a la disposición del maíz y se comparan con la clasificación visual de un humano para comprobar la eficiencia del robot. Las pruebas fueron realizadas en el mes de diciembre, bajo condición invernal en horarios de 9 a 12 y de 14 a 17 horas, donde el procesamiento era mejor.

c. Deshierba sobre V7

El maíz en esta etapa se encuentra con una altura promedio de 60cm con respecto al suelo, la prueba se realizó sobre 166 plantas que estuvieron dispuestas sobre cuatro hileras de cultivo (éstas son dos trayectorias para cubrir con el robot), en total se contabilizaron 8 pruebas de acuerdo a la disponibilidad del cultivo, la distribución de la mala hierba es totalmente aleatoria por lo que hay áreas de trabajo vacías (sin presencia de mala hierba), entonces se realiza el análisis de clasificación del maíz en la imagen y de las malas hierbas en el área de trabajo del robot delta, como se muestra a continuación en la tabla 34:

Tabla 34
Pruebas de clasificación para Maíz V7

Pruebas de clasificación humana y de Raspberry Pi						
Prueba N°	Áreas de trabajo (Muestras)	Plantas maíz	Clasificación Humana		Deshierba Robot	
			Maíz	Mala Hierba	Maíz	Mala Hierba
1	18	22	55	24	51	28
2	19	22	54	40	50	35
3	5	10	17	12	20	28
4	12	16	40	24	38	23
5	21	28	62	35	65	38
6	19	24	55	31	52	33
7	13	18	39	19	40	20
8	27	26	81	42	73	53
Total	134	166	403	227	389	258

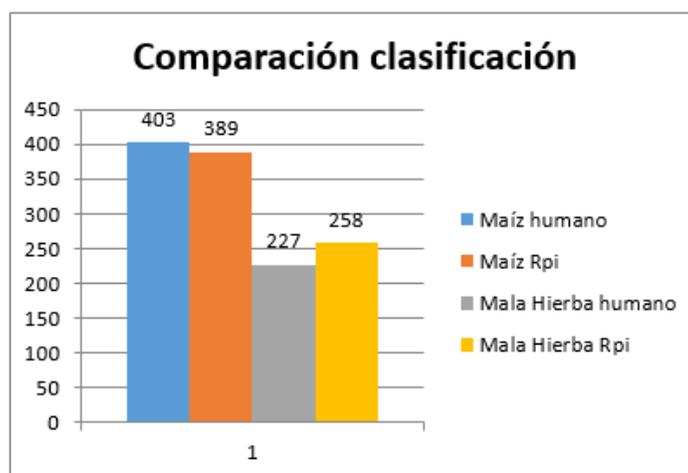


Figura 155 Comparación de la clasificación maíz V7

Cómo se puede apreciar en la tabla 34 y en la figura 155, hay 134 áreas de trabajo del robot delta, en ellas hubo 258 clasificaciones por el Raspberry Pi sobre mala hierba contra 227 clasificaciones por el clasificador humano, el resultado es razonable ya que el proceso de visión pudo haber generado más contornos por alguna condición ambiental puntal(luz, partículas de tierra), el resultado de clasificación de plantas fue más parejo, ya que el 96,52% de plantas fueron clasificadas por el microcomputador, con lo que se comprueba que para este desarrollo vegetativo del maíz, la clasificación es adecuada. Con los datos obtenidos de la clasificación de maíz, se puede realizar un análisis de que tan efectivo fue el proceso de deshierba, por lo que ahora se presenta un análisis de la deshierba sobre cada área de trabajo para el maíz de etapa V7.

Tabla 35
Pruebas de deshierba sobre áreas de trabajo de maíz V7

Pruebas deshierba para maíz V7					
Prueba N°	Áreas de trabajo	Áreas de trabajo desherbadas más del 70%	Áreas de trabajo desherbadas menos del 70%	Áreas sin mala hierba	Total deshierba
1	18	11	3	4	78,57%
2	19	12	6	0	66,67%
3	5	5	1	0	83,33%
4	12	9	3	0	75,00%
5	21	15	4	2	78,95%
6	19	14	3	2	82,35%
7	13	8	3	2	72,73%
8	27	18	6	3	75,00%
Total	134	92	29	13	$\bar{x} = 77\%$

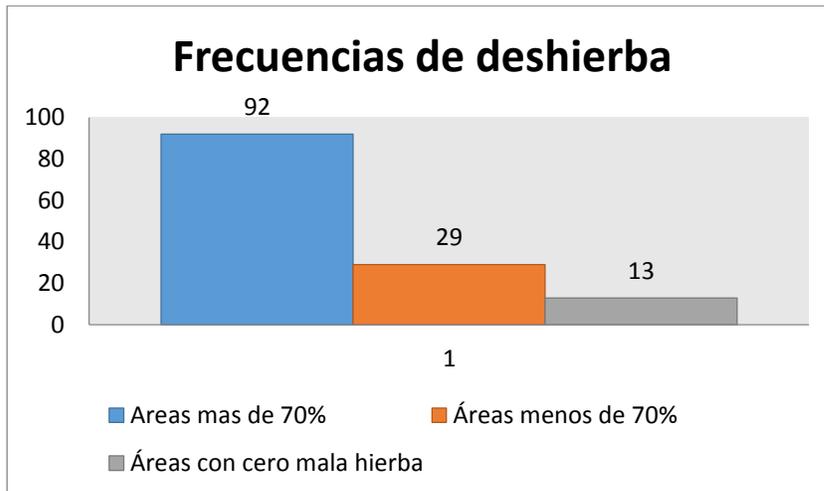


Figura 156 Frecuencia de deshierba en maíz V7

Para éste análisis se observó los datos de cada prueba de la deshierba en cada área de trabajo individual, se contabilizó las áreas de trabajo que cumplían con tres categorías: Las que cumplían con más del 70% de deshierba, menos del 70% y áreas sin mala hierbas. La media de los resultados de deshierba es aceptable para los objetivos de este proyecto, pues el porcentaje de acierto es del 77% del total de las áreas que contienen mala hierba, con lo que se concluye que el proceso es adecuado para este estado vegetativo del maíz.

d. Deshierba sobre V10

Para esta prueba, se considera esta etapa del maíz, que tiene un promedio de 90 cm de altura con respecto al suelo, al igual que en el caso anterior la prueba se realizó sobre 164 plantas que estuvieron dispuestas sobre cuatro hileras de cultivo (estás son dos trayectorias para cubrir con el robot), en total se contabilizaron 5 pruebas de acuerdo a la disponibilidad del cultivo, al igual que en la anterior prueba, la distribución de la mala hierba es totalmente aleatoria, entonces se realiza el análisis de clasificación del maíz en la imagen y de las malas hierbas en el área de trabajo del robot Delta, como se muestra a continuación:

Tabla 36
Pruebas de clasificación para Maíz V10

Pruebas de clasificación humana y de Raspberry Pi						
Prueba Nº	Áreas de trabajo	Plantas maíz	Clasificación Humana		Deshierba Robot	
			Maíz	Mala Hierba	Maíz	Mala Hierba
1	21	36	57	31	59	29
2	21	32	62	40	51	41
3	23	36	65	55	57	69
4	20	26	54	46	52	55
5	23	34	72	51	60	45
Total	108	164	310	223	279	239

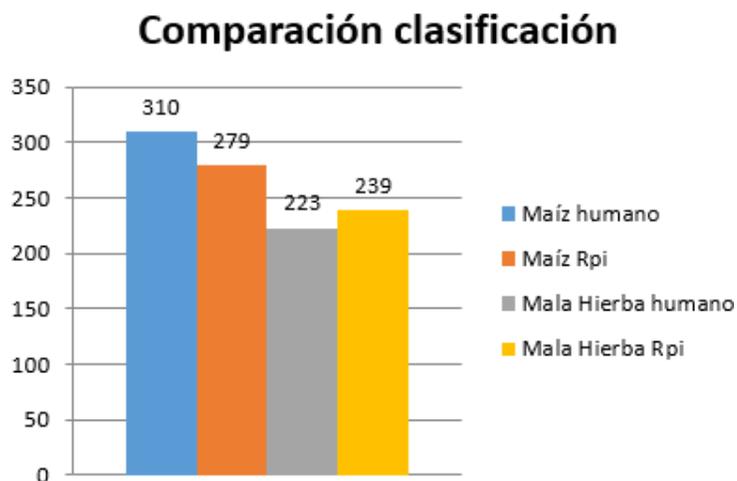


Figura 157 Comparación de la clasificación maíz V10

En esta prueba, la clasificación de maíz fue menos efectiva, principalmente debido a la altura del cultivo que en algunas ocasiones no encajaba en la imagen obtenida de la cámara, el porcentaje de detección del sistema esta vez fue del 90%. En el caso de las malas hierbas hay la misma condición que en la prueba

anterior, las malas hierbas detectadas por el sistema superan a las detectadas por el clasificador humano, por lo que hay más ciclos de trabajo en el robot delta debido a este fallo en la segmentación de algunas imágenes. Ahora se presenta el análisis de deshierba por áreas.

Tabla 37
Pruebas de deshierba sobre áreas de trabajo de maíz V10

Pruebas deshierba para maíz V10					
Prueba N°	Áreas de trabajo	Áreas de trabajo desherbadas más del 70%	Áreas de trabajo desherbadas menos del 70%	Áreas sin mala hierba	Total deshierba
1	21	9	3	9	75,00%
2	21	13	5	3	72,22%
3	23	17	5	2	77,27%
4	20	11	6	3	64,71%
5	23	11	6	5	64,71%
Total	108	61	25	22	$\bar{x} = 70,78\%$

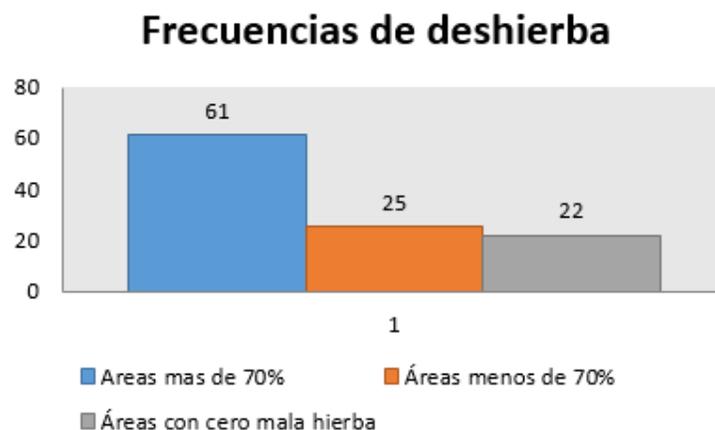


Figura 158 Frecuencia de deshierba en maíz V10

Para este caso, el crecimiento del maíz era muy notorio, en algunos casos casi se cerraba la hilera, lo que hace que algunas malas hierbas sean mal segmentadas o no se detecten por estar muy cerca de la hojas del maíz, principalmente a esto se atribuye que la media del porcentaje de deshierba haya llegado al 70%, que es todavía aceptable para el objetivo del presente documento.

5.3 Validación de la Hipótesis

Una vez finalizada la parte de los resultados, es necesario comprobar que la hipótesis planteada al inicio se cumpla, para lo cual se utiliza la prueba estadística de chi-cuadrado (también llamada ji cuadrado) para demostrar que los datos obtenidos no son resultados del azar.

5.3.1 Hipótesis nula y alternativa

Para usar la prueba de chi cuadrado, es necesario plantear dos hipótesis según (Navidi, 2006) a la primera se le llamara *hipótesis nula* (H_0) que establece que el efecto que indica la muestra es atribuible solamente a la variación aleatoria entre la muestra y la población. La *hipótesis alternativa* (H_1) establece que el efecto que indica la muestra es verdadero, ya que representa a toda la población, entonces se presentan las dos hipótesis para este proyecto.

- H_0 : El prototipo deshierba hasta el 70% del área de trabajo en las dos siembras de maíz
- H_1 : El prototipo desherbará más el 70% del área de trabajo en las dos siembras de maíz

5.3.2 Observaciones

Una vez que se han establecido las hipótesis, se incluye una tabla de observaciones que es la que tiene los valores obtenidos en las pruebas realizadas, estos valores incluyen las pruebas con maíz de etapa vegetativa V7 y V10 días (dos poblaciones distintas), la tabla es la siguiente.

Tabla 38
Tabla de valores observados

	Deshierba > 70%	Deshierba < 70%	Áreas sin mala hierba	Total
Maíz V7	92	29	13	134
Maíz V10	61	25	22	108
Total	153	54	35	242

5.3.3 Prueba de bondad de ajuste

Para los datos anteriormente descritos, se utiliza un nivel de significancia del 5% ($\alpha = 0.05$), la regla de decisión es que si el valor del estadístico de prueba es menor al valor crítico, este está en la zona de aceptación con lo que se concluye que la hipótesis nula es acertada, caso contrario se opta por la hipótesis alternativa, en términos matemáticos $\chi^2 \leq \chi^2_{\alpha}$. Entonces según (Navidi, 2006) se tiene que calcular un valor de frecuencias esperadas (E_{ij}) para cada celda, este valor se obtiene a través de la siguiente relación.

$$E_{ij} = \frac{O_i O_j}{O_{..}} \quad (64)$$

Dónde:

O_i : Es el valor del total de la fila de la tabla

O_j : Es el valor del total de la columna de la tabla

$O_{..}$: Es el valor del total de todas las muestras de la tabla

Conociendo la relación, entonces se calcula una tabla de valores esperados para la distribución del prototipo, con lo que queda de la siguiente forma

Tabla 39
Distribución de valores esperados

	Deshierba>70%	Deshierba<70%	Áreas sin mala hierba	Total
Maíz V7	84,72	29,90	19,38	134,00
Maíz V10	68,28	24,10	15,62	108,00
Total	153,00	54,00	35,00	242,00

Cómo se puede verificar, todos los valores observados son mayores a 5, por lo que la prueba de chi cuadrado es adecuada. Falta calcular la diferencia de los valores observados con los esperados, como (Navidi, 2006) menciona en la siguiente fórmula.

$$\chi^2 = \sum_{i=1}^I \sum_{j=1}^J \frac{(O_{ij} - E_{ij})^2}{E_{ij}} \quad (65)$$

Una vez calculado el estadístico de prueba, queda de la siguiente forma:

$$\chi^2 = 6,17$$

5.3.4 Grados de libertad

(Navidi, 2006) además, menciona que se debe obtener el grado de libertad de la prueba, pues con este valor se podrá comparar los valores dados con el valor que se encuentra en las tablas de la distribución de chi cuadrado, entonces el autor refiere a la siguiente ecuación:

$$gdl = (M - 1)(N - 1) \quad (66)$$

Como se tiene dos filas y tres columnas, el número de grados de libertad para el cálculo es $gdl = (2 - 1)(3 - 1) = 2$. Con lo que se obtiene dos grados de

libertad, la significancia para $\alpha = 0,05$ es la siguiente $\chi^2_{\alpha} = 5,991$, entonces se compara con el estadístico de prueba:

$$\chi^2 \leq \chi^2_{\alpha}$$

$$6,17 \leq 5,991$$

No se cumple

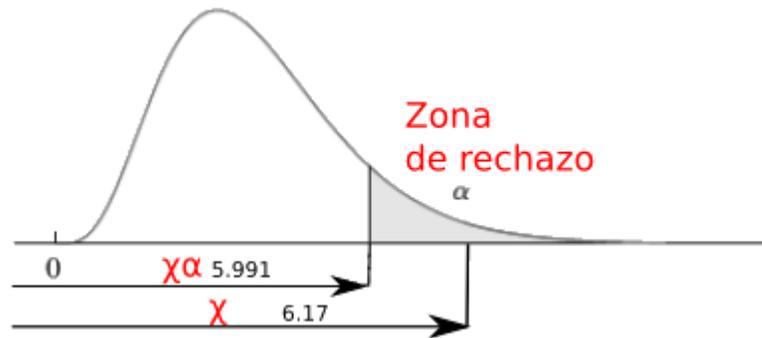


Figura 159 Distribución de Chi Cuadrado con los valores obtenidos

Cómo se observa en la relación anterior y en la anterior figura, el valor se encuentra en la zona de rechazo, por lo que hay evidencia para rechazar la hipótesis nula y se queda con la alternativa, con lo que se comprueba que las pruebas realizadas no fueron resultado del azar sino que son viables para cualquier tipo de cultivo de maíz en zonas planas.

5.4 Limitaciones

- Terreno muy irregular
- Distribución de mala hierba no diferenciable(muy cercanas)
- Terrenos muy sueltos
- Ancho de hilera de maíz menor a 800mm
- Etapas vegetativas superiores a la V10
- Consistencia del terreno muy dura
- Horas con mucha o muy poca iluminación solar

5.5 Costos

Para la elaboración del presente proyecto fue necesaria la adquisición de diferentes componentes y la fabricación de otros. En la siguiente tabla se detalla todos estos valores.

Tabla 40

Descripción de costos del desarrollo del prototipo

Descripción	Cant.	Costo Unitario \$	Costo Total \$
Rotulas esféricas	12	1,23	14,76
Varilla Roscada m4	3	3,85	11,55
Varilla roscada m3	1	4	4
Tuercas de presión m3	12	0,1	1,20
Tubo Aluminio 5/16	1	4	4
Fragmento de plancha de Aluminio	2	9	18
Corte Plasma	1	5	5
Trozo de Perfil Aluminio tipo C	1	3	3
Servomotores Tower Pro MG996R	3	12,50	37,50
Impresión 3D Base Inferior Delta	1	13	13
Impresión 3D Herramienta de corte	1	3	3
Motor Herramienta de Corte	1	8	8
Reductor Motor de Herramienta	1	6	6
Tubo cuadrado de acero 1 in	3	9	27
Tubo de acero de acero $\frac{3}{4}$ x2m	1	3	3
Plancha de acero galvanizado 0.7mm	1	19	19

CONTINÚA 

Ruedas de Carretilla	2	9	18
Rueda Loca	1	36	36
Suspensión KeZhen 750	2	11,50	23
Platinas	3	1,5	4,50
Suelda	1	50	50
Motores Banebots RS775 18V	2	36,5	73
Caja Reductora 152,88:1 Banebots	2	120	240
Tarjeta Cytron 30 ^a	2	64,50	129
Rueda para encoder	2	2	4
Tarjeta Raspberry PI 3	1	85	85
Módulo Raspberry Pi Camera V2	1	25	25
Sensores tipo Herradura	2	0,65	1,3
Batería sbb AGM 20Ah/12VDC	2	91,20	182,40
Banco de Baterías ARGOMTech 10000mAh	1	18,60	18,60
Arduino Uno	1	23	23
Luces Piloto, paro de Emergencia	-	6	6
Cables, tornillos y otros	-	-	30
Pintura	-	45	45
Costos por desarrollo	1	1000	1000
Total			\$ 2170,61

El costo total del proyecto llega al valor de 2170,61 dólares, cabe destacar que en los valores de algunos elementos se le sumo el costo de importación y aun así el costo es menor a robots con la misma funcionalidad como el ecoRobotix, el cual tiene un costo de 15000 euros (\$17850) (Robophil, 2017).

CAPÍTULO VI

CONCLUSIONES Y RECOMENDACIONES

6.1 Conclusiones

- Se diseñó y construyó un prototipo robotizado de deshierba de cultivos de maíz en su etapa inicial, el cual cuenta con: una plataforma móvil de configuración diferencial, un robot paralelo tipo Delta el cual controla la posición de la herramienta de eliminación de mala hierba, un sistema de suspensión, un sistema de visión artificial y un panel de control externo para las condiciones del terreno de cultivo de maíz en su etapa inicial.
- Se diseñó y construyó un robot paralelo tipo Delta, en base a un área de trabajo establecido limitado a las características de la siembra del cultivo de maíz según el estándar del SINAGAP, el cual es capaz de controlar el posicionamiento con bajo error de la herramienta encargada de eliminar las malas hierbas en los campos de maíz.
- La configuración diferencial fue la elegida para el prototipo debido a sus múltiples aplicaciones en robótica móvil, principalmente por su sencillez de uso y que no necesita de sensores muy sofisticados, esta configuración sufre por distintos factores como la desalineación de las ruedas, la variación de ancho de las mismas y el deslizamiento pero es mejorable al usar otros métodos como GPS con acelerómetros.
- En el prototipo fue necesario la implementación de un sistema de suspensión, debido a que el terreno de trabajo para la deshierba no es regular, con esto se consigue mejorar la estabilidad así como minimizar el deslizamiento de las llantas para así evitar fallos en la navegación.
- La implementación de una plataforma móvil que contiene un sistema de suspensión para atravesar terrenos de cultivo demostró un buen rendimiento a la hora de navegar en zonas planas de hileras de cultivos de maíz, la plataforma permite no dañar las plantas de maíz, además que transporta de manera segura a los elementos de control y tiene poco

deslizamiento que permite mantener en términos aceptables el error de navegación al usar la técnica de odometría por el uso de encoders.

- El microcomputador Raspberry Pi muestra un buen tiempo de procesamiento de imágenes, sin embargo está lejos de realizar un procesamiento y clasificación de una red neuronal convolucional en tiempo real, pero se consideró aceptable para el prototipo, según lo establecido en los objetivos de este proyecto.
- El sistema de visión mostró un adecuado procesamiento para el prototipo, con un proceso de filtrado de la imagen que se obtiene de la cámara se puede obtener una distribución compacta de los contornos, que luego permitirá realizar una adecuada clasificación de las plantas, generar las hileras y enviar los datos al robot Delta para malas hierbas mayores a 30 mm² de área. El proceso sufre también de debilidades al presentan luces muy fuertes en las hojas o sombras, además que las partículas de suelo se pegan contra las hojas de las plantas más superficiales, con lo que hacen más difícil su detección.
- El modelo vigente de Machine Learning en la actualidad son las Redes Neuronales Convolucionales(CNN), su alta efectividad hacen su uso obligatorio en aplicaciones donde la precisión sea crítica, el modelo cNET es recomendado para aplicaciones de clasificación de plantas con un 97% de acierto en el caso de este proyecto, las dos únicas desventajas de este tipo de modelos es que necesitan de muchas muestras para el entrenamiento y el uso de dispositivos especializados en CNN para un tiempo de procesamiento más eficiente.
- Para la comunicación entre las dos tarjetas principales se implementó un protocolo de comunicación tipo Serial RS232 a una velocidad de transferencia de datos de 115200 Baudios, lo cual permite una transmisión rápida y segura, al usar cable USB blindado.
- Se realizaron pruebas de funcionamiento en un campo de maíz, que contenía dos siembras con plantas correspondientes a las etapas vegetativas V7 y V10, en el lado de la clasificación, el sistema demostró

que para la etapa V7 se obtuvo un 96% de clasificación acertada de plantas de maíz y para V10 de un 90%, estos valores son bastante acertados para cuando el funcionamiento se da en etapas de maíz inferiores al estado V7, debido a que en V10, las plantas son muy grandes y algunas salen del campo de visión de la cámara.

- En la misma prueba, se realizó la deshierba sobre 134 áreas de trabajo de maíz de etapa V7 y 108 áreas de trabajo de maíz de etapa V8, los resultados fueron que en la primera etapa mencionada, se logró un 79% de deshierba y para la otra etapa se logró un 70%, las cuales se encuentran más arriba del valor que se propuso en este proyecto.
- El error que se obtiene en la deshierba se atribuye a múltiples factores como la mala segmentación porque hay plantas que se interponen sobre otras y distribución muy cercana de mala hierba, además del error de posicionamiento de navegación de la plataforma móvil. Sin embargo por el diámetro de la herramienta aún si existe error de posicionamiento del brazo Delta se logra al menor dañar en parte a la planta no benéfica, con lo que se evitaría su completo desarrollo.

6.2 Recomendaciones

- Para una mejor detección de la mala hierba se recomienda investigar sobre la espectroscopia y el contexto espacial (Ahmad, Tang, & Steward, 2014) para una mejor segmentación entre las plantas.
- Analizar la posibilidad de utilizar cámaras multiespectrales o infrarrojas para la operación de noche para robots agrícolas, pues la luz incide mucho en el procesamiento, con lo que en la noche se tendría condiciones más controladas
- Iniciar la captura de imágenes antes de que se presione el botón inicio porque la Pi Camera V2 tiene control automático de exposición, balance de blancos y niveles de negro, para activarse necesita un tiempo antes de capturar con lo que se logra una mejor imagen para el procesamiento,

además si se espera un pulso de inicio, asegurarse de liberar el buffer de la cámara antes de capturar otra imagen.

- Si se requiere una mayor potencia de procesamiento se recomienda usar una Unidad de Procesamiento Neural (NPU) adicional o una tarjeta compatible con Nvidia CUDA.
- Se recomienda usar una computadora con GPU para tener menores tiempos de entrenamiento y prueba para el desarrollo de la red
- Se debe dejar un adecuado espacio para el movimiento del robot tipo Delta, para evitar colisiones entre los otros elementos del prototipo y por ende evadir daños en el mismo,
- Se debe restringir los valores que ingresan la cinemática inversa del robot tipo Delta, puesto que al ingresar datos erróneos, pueden provocar colisiones entre el brazo Delta y la estructura del prototipo, o a su vez con el suelo, debido a que el valor del alcance máximo del eje Z es mayor que la distancia del brazo Delta al piso.
- Comprobar la efectividad de otros métodos para la eliminación de malas hierbas a ser implementadas en el robot delta como el método químico, térmico o eléctrico.
- Investigar el uso de configuraciones Skid Steer, Ackerman, Oruga para robots agrícolas pues ayudarían a reducir el deslizamiento en las ruedas y el error de posicionamiento.
- Se recomienda verificar que las llantas estén bien alineadas para un mejor desempeño de la navegación.
- Se debe procurar que la distancia entre las ruedas pueda ser ajustada y así poder adaptarse a distintos cultivos de maíz, que tengan distancia entre hileras de maíz distintas a las recomendadas de 800 mm.
- Para seleccionar el tipo de sensores para los encoders, se debe tomar en cuenta los efectos de la luz sobre el sensor.
- Se recomienda para la navegación comprar encoders (magnéticos o de otro tipo de principio de funcionamiento) de alta resolución y que de ser posible

vengan con los motores del robot, para así evitar errores sistemáticos en la navegación.

- Investigar otros métodos de Dead Reckoning como el uso de acelerómetros, magnetómetros o GPS que tendrán un menor error por deslizamiento que en el caso de los encoders.
- Usar estructuras en la programación permite entender de mejor forma el código, con lo que se puede agrupar muchos parámetros de un objeto, como en el caso de este proyecto las características del maíz, por lo que es muy recomendable su uso.
- El uso de librerías facilita la elaboración y prueba del código, puesto que cada librería puede ocuparse en distintos sistemas como un PC y el Raspberry Pi, aparte que sus cambios son más rápidos porque son porciones dedicadas de código.
- Para la realización del cableado evitar que los cables de los motores estén juntos con los de control, puesto que se generan errores por inducción magnética.
- Al realizar la comunicación entre la tarjeta Arduino y el Raspberry Pi se deben realizar confirmaciones de la información transmitida así como de cuando un proceso comienza o finaliza, además de incluir el carácter '\0' para que el otro dispositivo de comunicación sepa cuando ha finalizado la transmisión de datos, y así las tarjetas funcionen de manera sincronizada.
- Es recomendable usar el puerto USB del Raspberry Pi para conectar al puerto serial del Arduino, debido a que las conexiones del cable son seguras y blindadas para ruido eléctrico, además que el puerto `/dev/ttyACM0` permite velocidades hasta 115200 bauds.
- Al momento de poner en funcionamiento el prototipo, se debe asegurar que el terreno no se encuentre mojado o este muy blando puesto que esto dificulta la navegación enormemente.
- Pintar los elementos mecánicos una vez que se haya acabado de construir para evitar la corrosión de sus piezas.

REFERENCIAS BIBLIOGRÁFICAS

- Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., . . . Zheng, X. (2015). *TensorFlow: Large-scale machine learning on heterogeneous systems*. Recuperado el 01 de Mayo de 2017, de www.tensorflow.org
- Ahmad, T., Tang, L., & Steward, B. (2014). Automated Mechanical Weeding. En S. Young, & F. Pierce, *Automation: The Future of Weed Control* (págs. 125-137). Dordrecht: Springer Science+Business Media.
- Ampflow. (23 de Marzo de 2017). *Ampflow Pancake Motos*. Recuperado el 23 de Marzo de 2017, de Ampflow: http://www.ampflow.com/pancake_motors.htm
- Arduino. (12 de Abril de 2017). *Arduino Uno Rev3*. Recuperado el 12 de Junio de 2017, de <https://store.arduino.cc/usa/arduino-uno-rev3>
- ARGOMTech. (15 de Julio de 2017). *Cell Phone Power Bank 10000 mAh 1A/2.1A*. Recuperado el 15 de Julio de 2017, de <https://www.argomtech.com/products/cell-powerbank-10000-mah-arg-ac-0239>
- ASOTV. (03 de Agosto de 2017). *As seen of TV*. Recuperado el 03 de Agosto de 2017, de <http://www.asseenontv.com/ultimate-auger-makes-yard-work-easy/detail.php?p=296689>
- Balmaceda, A. (2011). *Metodología de rediseño de un robot paralelo tipo delta de 3 grados de libertad en función de un espacio de trabajo prescrito*. Querétaro: s.e.
- Banebots. (24 de Marzo de 2017). *RS775 Motor - 18V*. Recuperado el 24 de Marzo de 2017, de Banebots: <http://www.banebots.com/product/M7-RS775-18.html>
- Banggood. (17 de Julio de 2017). *Traxxas Rod Ends*. Recuperado el 17 de Julio de 2017, de Traxxas Rod Ends: <https://www.banggood.com/Traxxas-Rod-Ends-wHollow-Balls-Large-Revo-5347-NIB-M4-For-3D-Printer-p-1026300.html>
- Beagleboard.org. (23 de Marzo de 2017). *BeagleBone Black*. Recuperado el 25 de Junio de 2017, de <https://beagleboard.org/black>
- Blanco, J. (13 de Octubre de 2013). *MRTP Empowering C++ development in robotics*. Recuperado el 19 de Mayo de 2017, de https://www.mrpt.org/tutorials/programming/odometry-and-motion-models/probabilistic_motion_models/
- Bolton, W. (2011). *Mecatrónica sistemas de control electrónico en ingeniería mecánica y eléctrica*. Michigan: Alfaomega.
- Brahmbhatt, S. (2013). *Practical OpenCV*. New York: Springer Science Business Media New York.

- Brinder, M., Hirokawa, N., & Windhorst, U. (2009). *Encyclopedia of Neuroscience*. Heidelberg: Springer Berlin Heidelberg.
- Brivot, R., & Marchant, J. (1996). Segmentation of plants and weeds for a precision cop protection robot using infrared images. *IEEE Proc.- Vis. Image Signal Process.*
- Budabar. (2012 de Marzo de 2015). *Depositphotos*. Recuperado el 18 de Marzo de 2017, de Depositphotos: <https://sp.depositphotos.com/48108339/stock-photo-weeding-corn-field-with-hoe.html>
- Bueno, G., Suarez, O., Espinoza, J., Salido, J., Serrano, I., & Noelia, V. (2015). *Learning Image Processing with OpenCV*. Birmingham: Packt Publishing Ltd.
- Call, B., Malespina, G., Varela, E., & Palomeque, C. (2007). *Detección y Seguimiento de objetos con OpenCV*. Sarmiento: Universidad Tecnológica Nacional.
- Carrod Electrónica. (Octubre de 2017). *Sensor FC03 Detector de Velocidad para Motor Tipo Herradura Breakout*. Recuperado el 23 de Octubre de 2017, de <https://www.carrod.mx/products/sensor-fc03-detector-de-velocidad-para-motor-tipo-herradura-breakout>
- Cirelli, G. (14 de Febrero de 2015). *PanamaHitec*. Recuperado el 19 de Junio de 2017, de <http://panamahitek.com/servomotor-con-acelerometro-mma7361/>
- Coleman, B. (13 de Marzo de 2017). *Robotshop*. Recuperado el 29 de Marzo de 2017, de <http://www.robotshop.com/blog/en/how-to-make-a-robot-lesson-5-motor-controller-3695>
- Cook, D. (2015). *Robot Building for Beginners*. New York: Apress, Berkeley, CA.
- Corporación Supertren. (2017). *Harris Autoparts*. Recuperado el 03 de Julio de 2017, de Corporación Supertren MACA: <http://corporacionsupertren.com/st/>
- Crespo, J. (6 de Julio de 2016). *Aprendiendo Arduino*. Recuperado el 6 de 2017 de Julio, de Motor DC: <https://aprendiendoarduino.wordpress.com/tag/motor-dc/>
- CYTRON. (28 de Marzo de 2017). *Cytron 30A DC Motor Driver*. Recuperado el 26 de Julio de 2017, de <https://www.cytron.io/p-md30c>
- Deshpande, A. (30 de Mayo de 2017). *A Beginner's Guide To Understanding Convolutional Neural Networks*. Recuperado el 30 de Mayo de 2017, de <https://adeshpande3.github.io/adeshpande3.github.io/The-9-Deep-Learning-Papers-You-Need-To-Know-About.html>
- Díaz del Castillo, F. (15 de Abril de 2017). "TABLAS Y GRAFICAS PARA DISEÑO DE ELEMENTOS DE MÁQUINAS". Recuperado el 15 de Abril de 2017, de

http://olimpia.cuautitlan2.unam.mx/pagina_ingenieria/mecanica/mat/mat_mec/m1/tablas%20y%20graficas.pdf

- Drill Till. (3 de Agosto de 2017). Recuperado el 12 de Agosto de 2017, de Drill Till:
<https://www.drilltill.com/products/drill-till>
- Ecorobotix. (18 de Marzo de 2017). *Ecorobotix*. Recuperado el 18 de Marzo de 2017, de
<http://www.ecorobotix.com/old/category/project/>
- Ecured. (2017). *Microcontroladores*. Recuperado el 18 de Septiembre de 2017, de Ecured:
<https://www.ecured.cu/Microcontrolador#Caracter.C3.ADsticas>
- Electric Bricks. (25 de Mayo de 2010). *Electric Bricks*. Recuperado el 5 de Julio de 2017, de Electric Bricks: <http://blog.electricbricks.com/2010/05/caster-wheel/>
- Ergonica. (03 de Agosto de 2017). *Weeder Features - Compare and Review Weeding Tools*. Recuperado el 03 de Agosto de 2017, de Tools classifications:
http://www.ergonica.net/weeder_features_p2.htm
- Ertbølle, T., & Lavdal, H. (2001). *Mobile robot for weeding*. Ballerup, Dinamarca: Departamento de Control y Diseño Ingenieril Universidad Técnica de Dinamarca.
- FAO. (2008). *El Maíz en la nutrición humana*. Recuperado el 12 de Julio de 2017, de Depósito de documentos de la FAO: <http://www.fao.org/docrep/t0395s/T0395S02.htm>
- Fassio, A., Carriquiry, A., Tojo, C., & Romero, R. (1998). *MAIZ, Aspectos sobre fenología*. Montevideo: Unidad de Difusión e Información Tecnológica.
- FireCloud Cycles. (2016). *Suspension Units*. Recuperado el 15 de Junio de 2017, de
<https://www.firecloudcycles.co.uk/coil-shock-rear-suspension-unit-750lbsin-ke-zhen-150mm-black-bicycle-new-5401-p.asp>
- Future Electronics. (11 de Septiembre de 2017). *What is a Microcontroller?* Recuperado el 11 de Septiembre de 2017, de
<https://www.futureelectronics.com/en/Microcontrollers/microcontrollers.aspx>
- Garzón, G. (2 de Junio de 2012). *El automóvil al desnudo*. Recuperado el 2 de Junio de 2017, de SUSPENSIONES VEHÍCULOS (PARTE III):
<http://jeroitim.blogspot.com/2012/06/suspensiones-vehiculos-parte-iii.html>
- Geek Factory. (Octubre de 2017). *TCRT5000 Sensor Óptico Reflectivo*. Recuperado el 11 de Octubre de 2017, de <https://www.geekfactory.mx/tienda/sensores/tcrt5000-sensor-optico-reflectivo/>
- González, M. (2004). *El Terreno*. Barcelona: ESIC.
- Habibi, H., & Jahani, H. (2017). *Guide to Convolutional Neural Networks*. España: Springer Nature.

- Henderson, G. (12 de Octubre de 2017). *Wiring Pi GPIO Interface library for the Raspberry Pi*. Recuperado el 12 de Octubre de 2017, de <http://wiringpi.com/>
- Hijazi, S., Kumar, R., & Rowen, C. (2015). *Cadence Design Systems*. Recuperado el 18 de Mayo de 2017, de Cadence Design Systems: https://ip.cadence.com/uploads/901/cnn_wp-pdf
- Hong, J., Lei, T., & Heping, Z. (2011). Robust Crop and Weed Segmentation under Uncontrolled. *Sensors 2011*, 6271-6273.
- HP Wizard. (22 de Marzo de 2017). *Tire friction and rolling resistance coefficients*. Recuperado el 22 de Marzo de 2017, de HP Wizard: <http://hpwizard.com/tire-friction-coefficient.html>
- ImageNet. (02 de Junio de 2017). *Image-Net*. Recuperado el 02 de Junio de 2017, de <http://imagenet.org/>
- INTI-UE. (03 de Marzo de 2017). *Fundamentos de Diseño Mecánico*. Recuperado el 03 de Marzo de 2017, de INTI-UE: http://www.ue-inti.gob.ar/pdf/PPTs/CORTIZO/1_Fundamentos_de_diseno/03_CoeficienteSeguridad.pdf
- Javidan, A. (14 de Julio de 2006). *WORKS*. Recuperado el 11 de Julio de 2017, de Choosing the correct spring rate for your vehicle: http://www.worksevo.com/Spring_Rates_1.pdf
- Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., . . . Darrell, T. (20 de Junio de 2014). *Caffe: Convolutional Architecture for Fast Feature Embedding*. Recuperado el 28 de Abril de 2017, de <http://caffe.berkeleyvision.org/>
- Lara, F. (2005). *Fundamentos de Redes Neuronales Artificiales*. Recuperado el 13 de Abril de 2017, de http://conceptos.sociales.unam.mx/conceptos_final/598trabajo.pdf
- LeCun, Y., Cortes, C., & Burges, C. J. (29 de Mayo de 2017). *THE MNIST DATABASE of handwritten digits*. Recuperado el 29 de Mayo de 2017, de <http://yann.lecun.com/exdb/mnist/>
- Lee, S. H., Chan, C. S., Mayo, S. J., & Remagnino, P. (2017). How deep learning extracts and learns leaf features for plant classification. *Pattern Recognition*, 5-6.
- Liu, X.-J., Wang, J., & Pritschow, G. (2005). A new family of spatial 3-DoF fully-parallel manipulators with high rotational capability. *Mechanism and Machine Theory Volumen 4, Issue 4*, 475-494. Obtenido de A new family of spatial 3-DoF fully-parallel manipulators with high rotational capability
- López, R. (07 de Mayo de 2014). *¿Qué es y cómo funciona "Deep Learning"?* Recuperado el 07 de Mayo de 2017, de <https://rubenlopezg.wordpress.com/2014/05/07/que-es-y-como-funciona-deep-learning/>
- Luque, P., Álvarez, D., & Vera, C. (2004). *Ingeniería del automóvil: sistemas y comportamiento dinámico*. Madrid: Paraninfo.

- Makerlab-electronics. (23 de Octubre de 2017). *Makerlab-electronics*. Recuperado el 23 de Octubre de 2017, de Makerlab-electronics: <https://www.makerlab-electronics.com/product/digital-servo-motor-mg996r-180-rotation/>
- MEC. (2003). *El sistema robótico*. Recuperado el 23 de Agosto de 2017, de Análisis del sistema. Unidades funcionales: http://platea.pntic.mec.es/vgonzale/cyr_0204/ctrl_rob/robotica/sistema.htm
- Microchip. (2001). *PIC16F87X Datasheet*. USA. Obtenido de <http://www3.gobiernodecanarias.org/medusa/ecoblog/ravgon/files/2013/05/Caracter%C3%ADsticas-Arduino.pdf>
- MIPI Alliance. (2014). *MIPI Camera Serial Interface 2 (MIPI CSI-2)*. Recuperado el 23 de Marzo de 2017, de MIPI ALLIANCE: <https://mipi.org/specifications/mipi-camera-serial-interface-2-mipi-csi-2>
- Montabone, S. (2010). *Beginning Digital Image Processing Using Free Tools for Photographers*. United States of America: Springer Science, Business Media.
- Moreno, F., Blanco, J., & González, J. (2007). An Efficient Closed-Form Solution to Probabilistic 6D. *Advanced Concepts for Intelligent Vision Systems 9th International Conference, ACIVS, 932-942*.
- Muñoz, V. (1997). Planificación de trayectorias para robots móviles. En V. Muñoz, *Planificación de trayectorias para robots móviles* (págs. 21-52). Málaga: Departamento de Ingeniería de Sistemas y Automática de la Universidad.
- Naïo Technologies. (2013). *Naïo Technologies*. Recuperado el 09 de Marzo de 2017, de <http://www.naio-technologies.com/en/agricultural-equipment/weeding-robot-oz/>
- Navidi, W. (2006). *Estadística para ingenieros*. Colorado: McGraw-Hill.
- Nivia, E. (Diciembre de 2003). *Eco Portal*. Recuperado el 19 de Marzo de 2017, de Eco Portal: http://www.ambiente-ecologico.com/ediciones/2003/087_02.2003/087_Publicaciones_EcoPortal.php3
- Nuñez, F. (2016). *Diseño de un sistema de reconocimiento automatico de matriculas de vehículos mediante una red neuronal convolucional*. Barcelona: Universitat oberta de Catalunya.
- NVIDIA. (30 de Mayo de 2017). *Nvidia Developer*. Recuperado el 30 de Mayo de 2017, de CUDA FAQ: <https://developer.nvidia.com/cuda-faq>
- Ollero, A. (2001). *Robótica: Manipuladores y Robots Móviles*. Barcelona: Marcombo.
- Pariante Málaga, C. G. (2014). *Diseño y simulación de perforadora hidráulica para pozos de agua*. Piura-Perú: Universidad de Piura.

- Peñafiel, J. (07 de Febrero de 2015). *El Universo*. Recuperado el 7 de Marzo de 2017, de El Universo: <http://www.eluniverso.com/noticias/2015/02/07/nota/4525996/drones-vuelan-ver-estado-cultivos>
- Pérez, E., Guillermo, G., & Ferreira, J. (2012). *Control de malas hierbas en el cultivo de faba granja*. Recuperado el 11 de Marzo de 2017, de Servicio Regional de Investigación y Desarrollo Agroalimentario: <http://www.serida.org/publicacionesdetalle.php?id=5156>
- Pinout XYZ. (13 de Noviembre de 2017). *Pinout de Raspberry Pi - Pi 1 - B+, Pi 2*. Recuperado el 13 de Noviembre de 2017, de <https://es.pinout.xyz/pinout/#>
- Potena, C., Nardi, D., & Pretto, A. (2016). Fast and accurate crop and weed identification with summarized train sets for precision agriculture. *International Conference on INtelligent Autonomous Systems*. Springer, 105-121.
- Promotec. (25 de Junio de 2017). *promotec*. Obtenido de <https://www.promotec.net/producto/servo-mg995/#>
- Psyma. (28 de Noviembre de 2017). *¿Cómo determinar el tamaño de una muestra?* Recuperado el Diciembre de 2017, de <http://www.psyma.com/company/news/message/como-determinar-el-tamano-de-una-muestra>
- Rallt. (08 de Septiembre de 2017). *Maíz Patrimonio de la Humanidad*. Recuperado el 08 de Septiembre de 2017, de <http://www.rallt.org/PUBLICACIONES/libro%20del%20maiz%20completo.pdf>
- Raspberry Pi Camera V2 Camera Module*. (23 de Marzo de 2017). Recuperado el 23 de Marzo de 2017, de RS Components: <http://uk.rs-online.com/web/p/video-modules/9132664/>
- Raspberry Pi Foundation. (2016). *Raspberry Pi Foundation*. Recuperado el 20 de Abril de 2017, de <https://www.raspberrypi.org/products/raspberry-pi-3-model-b/>
- Raspberry Pi Foundation. (23 de Marzo de 2017). *Camera Module V2*. Recuperado el 12 de Junio de 2017, de Raspberry Pi: <https://www.raspberrypi.org/products/camera-module-v2/>
- Raszavian, A., Azizpour, H., Sullivan, J., & Carlsson, S. (2014). Cnn features off-the-shelf: an astounding baseline for recognition. *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, 806-813.
- Ray, S. (6 de Octubre de 2015). *Understanding Support Vector Machine algorithm from examples (along with code)*. Recuperado el 6 de Octubre de 2017, de <https://www.analyticsvidhya.com/blog/2015/10/understaing-support-vector-machine-example-code/>
- Rice University. (2017). *Robot Control*. Recuperado el 12 de Octubre de 2017, de Basic Control Methods:

<https://www.clear.rice.edu/elec201/Book/control.html#SECTION00171100000000000000>
0

- Ridetech. (18 de Septiembre de 2015). *RIDETECH STRONGARM DIMENSIONS*. Recuperado el 29 de Marzo de 2017, de <http://www.ridetech.com/info/ridetech-strongarm-dimensions/>
- RidgeSoft. (2005). *Programming your robot to Navigate*. Recuperado el 05 de Octubre de 2017, de www.ridgesoft.com
- Robophil. (23 de Noviembre de 2017). *Ecorobotix Interview*. Recuperado el 23 de Noviembre de 2017, de <https://www.robophil.com/reviews/ecorobotix-interview/>
- RobotShop. (Marzo de 2017). *Sabertooth Dual 60A 6V-30V Regenerative Motor Driver*. Recuperado el 28 de Marzo de 2017, de <http://www.robotshop.com/en/sabertooth-dual-motor-driver.html>
- Rocha, A. (2013). *Diseño y simulación de un sistema de control para una suspensión activa de un automóvil*. México: Azcapotzalco.
- Roston, G., & Krotkov, E. (1991). *Dead Reckoning Navigation for Walking Robots*. Pennsylvania: The Robotics Institute Carnegie Mellon University.
- Ruggerone, E. (13 de Marzo de 2015). *Report ONU, i pesticidi sono dannosi e soprattutto inutili*. Recuperado el 15 de Marzo de 2017, de [notizie.it Ambiente: http://www.notizie.it/report-ONU-i-pesticidi-sono-dannosi-e-soprattutto-inutili/](http://www.notizie.it/report-ONU-i-pesticidi-sono-dannosi-e-soprattutto-inutili/)
- Salazar, J., & Pachacama, W. (2006). *Diseño y Construcción de un Prototipo de Robot Delta para Aplicaciones Pick & Place*. Quito: Escuela Politécnica Nacional.
- Sammur, C., & Webb, G. (2017). *Encyclopedia of Machine Learning*. Australia: Springer Nature.
- Sánchez, R. (2008). *Accionamientos y actuadores eléctricos*. Huelva: Universidad de Huelva.
- Servodatabase. (2012). *ServoDatabase.com*. Recuperado el 23 de Abril de 2017, de <http://servodatabase.com/servo/hitec/hs-475hb>
- Siegwart, R., Nourbakhsh, I., & Scaramuzza, D. (2011). *Introduction to Autonomous Mobile Robots*. Londres: MIT Press.
- SINAGAP. (2015). *Etapas de producción del maíz*. Recuperado el 6 de Marzo de 2017, de [Prácticas de manejo: http://sinagap.agricultura.gob.ec/infoproductor/maiz/ciclo_maiz/ciclo_produccion_maiz.pdf](http://sinagap.agricultura.gob.ec/infoproductor/maiz/ciclo_maiz/ciclo_produccion_maiz.pdf)
- Smola, A., & Vishwanathan, S. (29 de Mayo de 2008). *Introduction to Machine Learning*. Cambridge: Cambridge University Press. Obtenido de <http://alex.smola.org/drafts/thebook.pdf>

- SODIMAC. (2017). *Rueda Neumática para Carretilla*. Recuperado el 01 de Junio de 2017, de <http://www.sodimac.cl/sodimac-cl/product/2838478/Rueda-Neumatica-para-Carretilla/2838478>
- Solderspot. (Octubre de 2017). *NavBot: Version 1*. Recuperado el 11 de Octubre de 2017, de <https://solderspot.wordpress.com/>
- Solomon, C., & Toby, B. (2011). *Fundamental of digital image precessing*. India: John Wiley & Sons, Ltd.
- Suárez, R., Piñeiro, J., & Valladares, J. (2005). Distintos sistemas de escarda en maíz forrajero. *Producciones agroganaderas: Gestión eficiente y conservación del medio natural*, 657-664.
- Sunbright Power co. (2016). *High Rating Battery Products*. Recuperado el 06 de Agosto de 2017, de High Rating Battery: http://www.sbb-battery.com/Products_View.asp?ID=7
- Syma Toys. (21 de Julio de 2017). *Syma X8C Venture*. Recuperado el 21 de Julio de 2017, de <http://www.symatoys.com/goodshow/x8c-syma-x8c-venture.html>
- Szeliski, R. (2010). *Computer Vision: Algorithms and Applications*. Seattle: Springer.
- Tablada, C., & Torres, G. (2007). *Redes Neuronales Artificiales*. Recuperado el 18 de Mayo de 2017, de <http://www.famaf.unc.edu.ar/~revm/digital24-3/redes.pdf>
- Taller Autojesa. (12 de Julio de 2017). *Amortiguadores Monroe® Adventure*. Recuperado el 12 de Julio de 2017, de <http://www.autojesa.com/taller-automovil-salamanca/amortiguadores-suspension-monroe-monroe-adventure.php>
- TecBolivia. (2013). *Tecnología en Bolivia*. Recuperado el 15 de Agosto de 2017, de Introducción a la Plataforma "Pinguino": <http://tecbolivia.com/index.php/articulos-y-tutoriales-microcontroladores/23-introduccion-a-la-plataforma-qpinguinoq-arduino-con-pic>
- Tellaeche, A., Pajares, G., Burgos-Artizzu, X., & Ribeiro, A. (2011). A computer vision approach for weeds identification through Support Vector. *Applied Soft Computing* 11, 908-915.
- Torres, F., Pomares, J., Gil, P., Puente, S., & Aracil, R. (2002). *Robots y Sistemas Sensoriales*. Madrid: Pearson Education.
- Torres, L. (09 de Diciembre de 2010). *UNAL (Universidad Nacional de Colombia)*. Recuperado el 24 de Mayo de 2017, de Redes Neuronales: <http://disi.unal.edu.co/~lctorress/RedNeu/LiRna008.pdf>
- TrossenRobotics. (2017). *AX Series DYNAMIXELS*. Recuperado el 17 de Abril de 2017, de Dynamixel AX-12A Robot Actuator: <http://www.trossenrobotics.com/dynamixel-ax-12-robot-actuator.aspx>

- UNED. (2011). *Sistemas embebidos - Información de referencia*. Recuperado el 18 de Septiembre de 2017, de http://www.ieec.uned.es/investigacion/Dipseil/PAC/archivos/Informacion_de_referencia_ISE5_3_1.pdf
- UNFPA Ecuador. (Noviembre de 2014). *UNFPA Ecuador*. Recuperado el 18 de Abril de 2017, de www.unfpa.org.ec
- Universidad de Nevada. (31 de Mayo de 2017). *Lecture*. Recuperado el 31 de Mayo de 2017, de <https://www.cse.unr.edu/~bebis/MathMethods/NNs/lecture.pdf>
- Universidad de Valladolid. (2005). *Detección - Sensores*. Recuperado el 19 de Septiembre de 2017, de <http://www.isa.cie.uva.es/~maria/sensores.pdf>
- V4L2 BCM2835 Driver. (5 de Mayo de 2017). Recuperado el 5 de Mayo de 2017, de Linux Documentation: <https://github.com/raspberrypi/linux/blob/rpi-3.10.y/Documentation/video4linux/bcm2835-v4l2.txt>
- Veličković, P. (12 de Septiembre de 2017). *Deep learning for complete beginners: convolutional neural networks with keras*. Recuperado el 12 de Septiembre de 2017, de Cambridge Spark: <https://cambridgespark.com/content/tutorials/convolutional-neural-networks-with-keras/index.html>
- Wang, P., Meng, Z., & Mei, H. (2013). Path Recognition for Agricultural Robot Vision Navigation under Weed Environment. *7th International Conference on Computer and Computing Technologies in Agriculture (CCTA)*, 242-248.
- Weed Spinner LLC. (03 de Agosto de 2017). *Weed Spinner*. Recuperado el 03 de Agosto de 2017, de Weed Spinner weed removal tool spins out weeds in one whirling motion!: <http://www.weedspinner.com/>
- Yáñez , C., Zambrano, J., Caicedo, M., & Heredia, J. (2013). *Guía de Producción de Maíz para pequeños agricultores y agricultoras*. Quito - Ecuador: INIAP.
- zBattery. (2 de Mayo de 2017). *EB20-12 BB Battery 12v 20Ah Sealed Lead Acid Battery*. Recuperado el 5 de Junio de 2017, de <http://www.zbattery.com/B-B-12v-20ah-Sla-High-Cycle-Use-Sla-Battery>

ANEXOS



DEPARTAMENTO DE CIENCIAS DE LA ENERGÍA Y MECÁNICA

CARRERA DE INGENIERÍA MECATRÓNICA

CERTIFICACIÓN

Se certifica que el presente trabajo fue desarrollado por los señores: **MAURICIO DANIEL BARRENO BARRENO** y **JOSÉ MISAEL JÁCOME BARRIONUEVO**

En la ciudad de Latacunga, a los **15 días del mes de diciembre del 2017.**

Ing. Andrea Córdova

DIRECTORA DEL PROYECTO

Ing. Vicente Hallo

DIRECTOR DEL CARRERA

Dr. Rodrigo Vaca

SECRETARIO ACADÉMICO

Aprobado por: