

DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA
CARRERA DE INGENIERÍA EN ELECTRÓNICA E
INSTRUMENTACIÓN


**PROYECTO PREVIO A LA OBTENCIÓN DEL TÍTULO DE
INGENIERO EN ELECTRÓNICA E INSTRUMENTACIÓN:**

**“DETECCIÓN DE OBJETOS EN TIEMPO REAL A
TRAVÉS DE REDES NEURONALES”**

INTEGRANTES:

- **Pablo S. Aulestia A.**
- **Jonathan S. Talahua R.**






**“EL SECRETO DEL ÉXITO NO RADICA EN FORTALECER
NUESTRAS DEBILIDADES, SINO EN POTENCIAR
NUESTRAS FORTALEZAS”.**

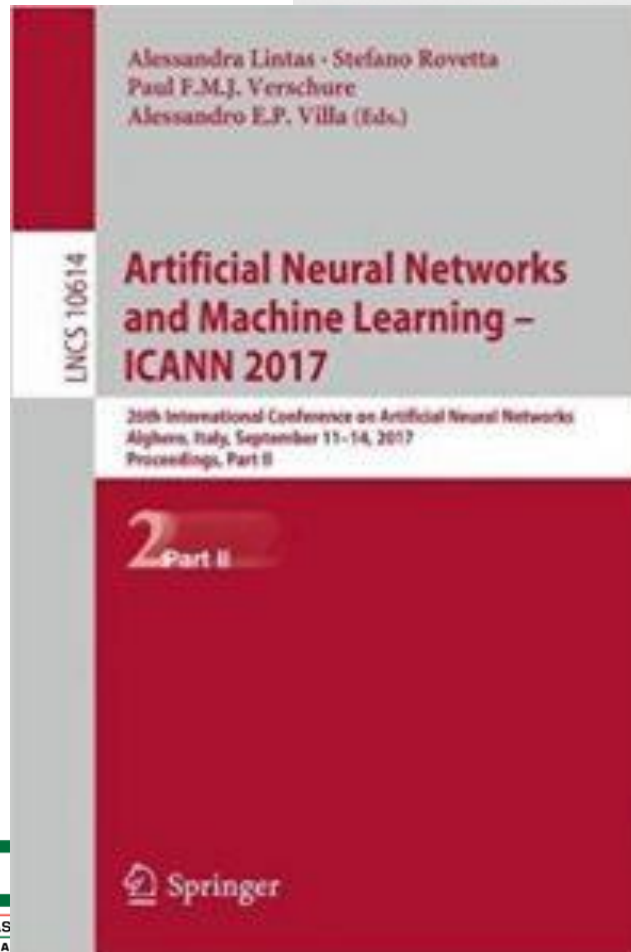
David Fischman

ARTÍCULO PUBLICADO EN: ICANN 2017, PART II, LNCS 10614

DOI: 10.1007/978-3-319-68612-7_67

 Springer Link

Search 




[International Conference on Artificial Neural Networks](#)

ICANN 2017: [Artificial Neural Networks and Machine Learning - ICANN 2017](#) pp 590-599 | [Cite as](#)

Real-Time Face Detection Using Artificial Neural Networks

Authors

[Authors and affiliations](#)

Pablo S. Aulestia, Jonathan S. Talahua, Víctor H. Andaluz, Marco E. Benalcázar 

Conference paper

First Online: 25 October 2017

Part of the [Lecture Notes in Computer Science](#) book series (LNCS, volume 10614)

Abstract

In this paper, we propose a model for face detection that works in both real-time and unstructured environments. For feature extraction, we applied the HOG (Histograms of Oriented Gradients) technique in a canonical window. For classification, we used a feed-



El Desafío de Lograr la Visión e Inteligencia Artificial

Por favor, diga en voz alta el resultado de las siguientes operaciones:

$$1 + 1$$



El Desafío de Lograr la Visión e Inteligencia Artificial

Por favor, diga en voz alta el resultado de las siguientes operaciones:

$$1 + 1$$

$$9 * 100$$



El Desafío de Lograr la Visión e Inteligencia Artificial

Por favor, diga en voz alta el resultado de las siguientes operaciones:

$$1 + 1$$

$$9 * 100$$

$$2^5$$



El Desafío de Lograr la Visión e Inteligencia Artificial

Por favor, diga en voz alta el resultado de las siguientes operaciones:

$$1 + 1$$

$$9 * 100$$

$$2^5$$

$$3.141516 * 1.25$$



El Desafío de Lograr la Visión e Inteligencia Artificial

Por favor, diga en voz alta el resultado de las siguientes operaciones:

$$1 + 1$$

$$9 * 100$$

$$2^5$$

$$3.141516 * 1.25$$

$$\sqrt{122.7} / 23$$



El Desafío de Lograr la Visión e Inteligencia Artificial

Por favor, diga en voz alta el resultado de las siguientes operaciones:

$$1 + 1$$

$$9 * 100$$

$$2^5$$

$$3.141516 * 1.25$$

$$\sqrt{122.7} / 23$$

A nuestro cerebro le cuesta mucho trabajo hacer estos cálculos, pero el computador resuelve estas operaciones muy rápidamente



El Desafío de Lograr la Visión e Inteligencia Artificial

Por favor, diga en voz alta lo que visualiza en cada imagen:



Nuestro cerebro es capaz de distinguir rápidamente el objeto que hay en cada imagen, pero esta tarea le cuesta mucho trabajo a un computador



¿Cómo Hacemos que el Computador Aprenda a Clasificar Imágenes con Autos de Imágenes con Motocicletas?

(Machine Learning - Entrenamiento)

➤ Necesitamos "varias" instancias de cada clase.



➤ Necesitamos modelo de clasificación (redes neuronales artificiales, SVM, kNN, regresión logística, etc.) y un algoritmo de aprendizaje (descenso de gradiente, etc.)

OBJETIVOS DEL PROYECTO:

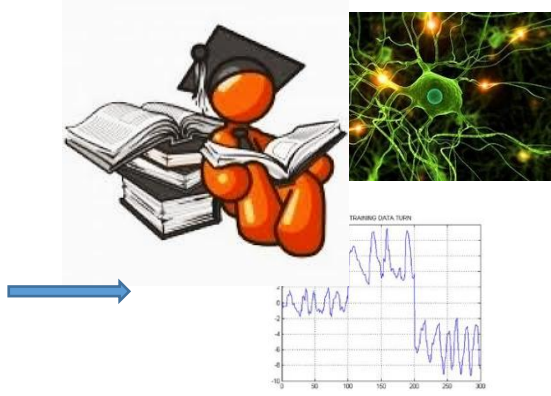
GENERAL:

• DISEÑAR UN ALGORITMO DE VISIÓN ARTIFICIAL



ESPECÍFICOS:

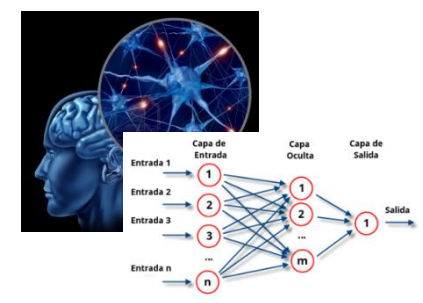
• INVESTIGAR LA FUNDAMENTACIÓN TEÓRICA



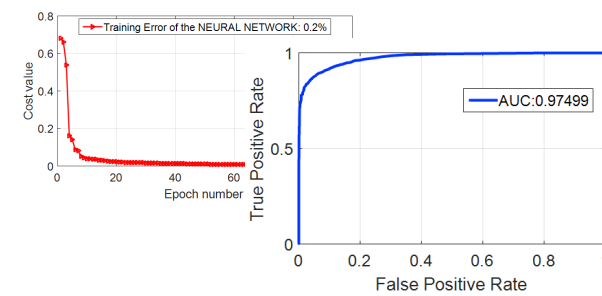
• EXTRAER LAS CARACTERÍSTICAS DE IMAGEN DE UN OBJETO PATRÓN



• DEFINIR LA ARQUITECTURA DE LA RED NEURONAL



• EVALUAR EXPERIMENTALMENTE EL FUNCIONAMIENTO



METODOLOGÍA DEL ALGORITMO

IMAGEN DE ENTRADA



Extracción de Características

Generación de Candidatos

Clasificación de Candidatos

Refinación del Sistema



METODOLOGÍA

Extracción de Características

Generación de Candidatos

Clasificación de Candidatos

Refinación del Sistema





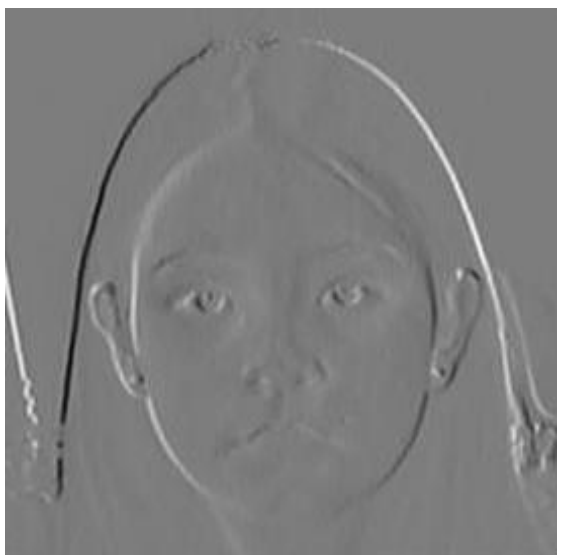
(Histograms of Oriented Gradients - HOG)

Img. Entrada
Esc. Grises



Patch I

Gradiente
horizontal (dx)



$$dx = I(x + 1, y) - I(x - 1, y)$$

Gradiente
vertical (dy)



$$dy = I(x, y + 1) - I(x, y - 1)$$

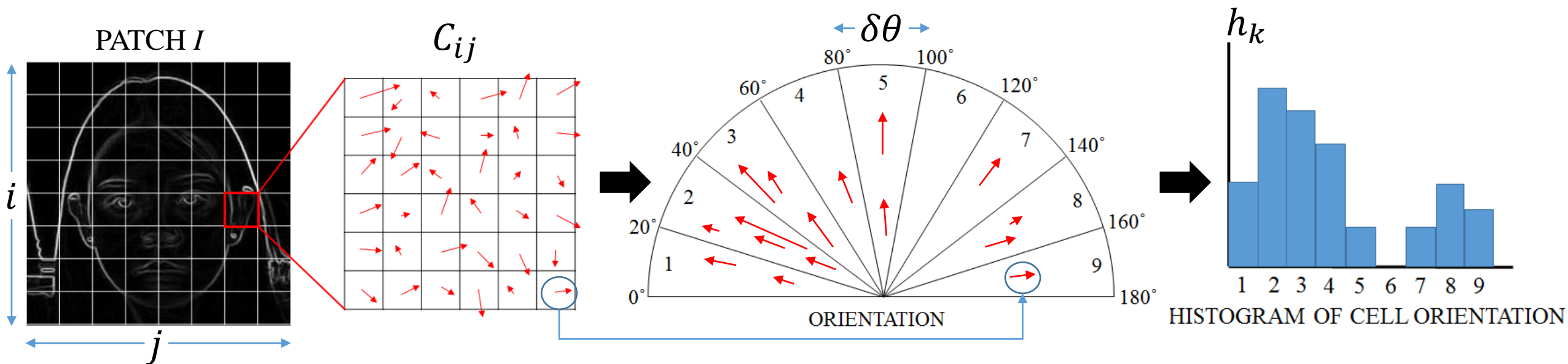
Magnitud
global



$$G(x, y) = \sqrt{dx^2 + dy^2}$$

$$\theta(x, y) = \arctan\left(\frac{dy}{dx}\right)$$

CÁLCULO DEL HISTOGRAMA



(a)

(b)

(c)

Dividimos el Patch I en celdas de 8x8 pixeles.

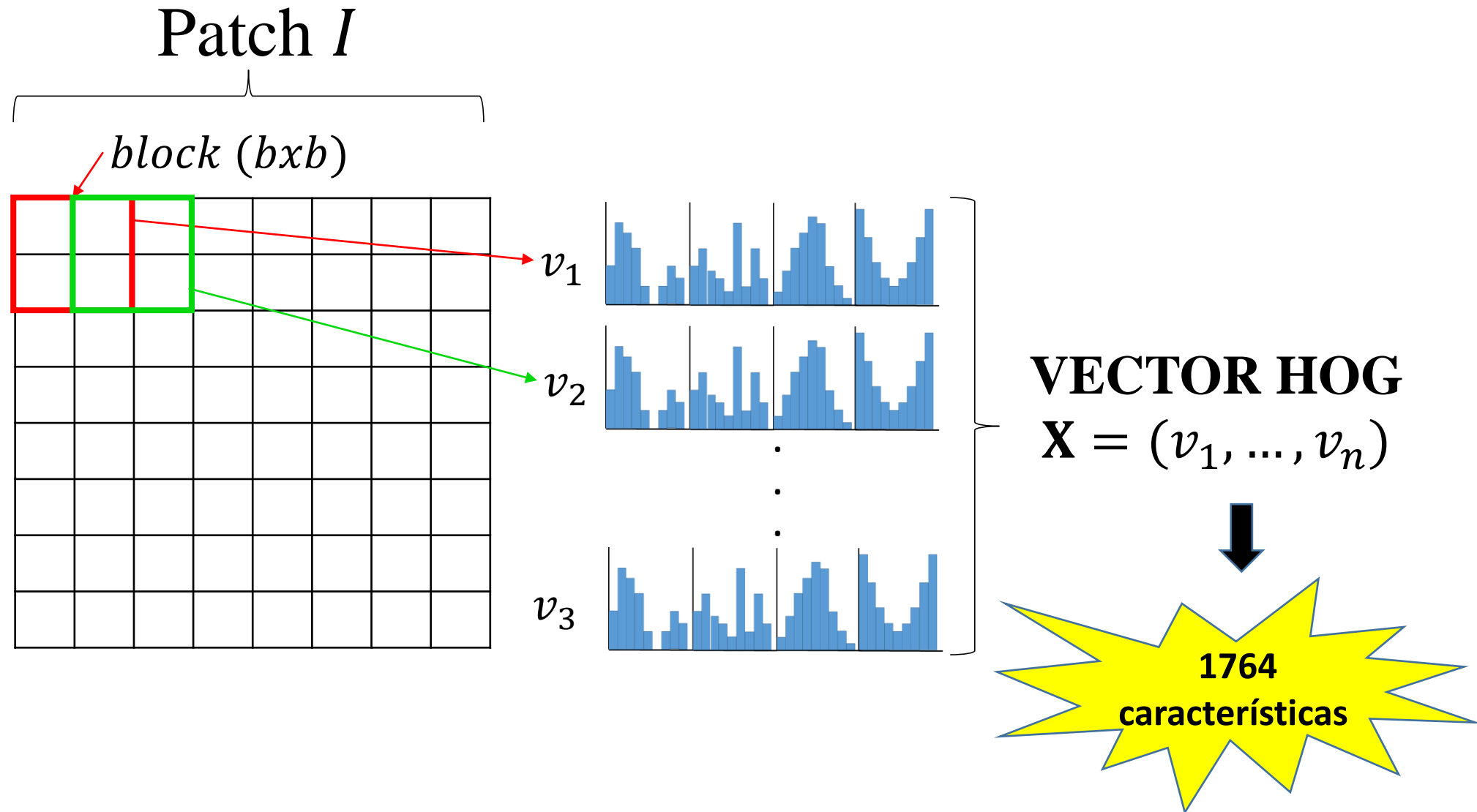
Definimos 9 intervalos de orientación entre 0° y 180° .

Cada celda debe ser asignada a un intervalo de orientación para generar su histograma.



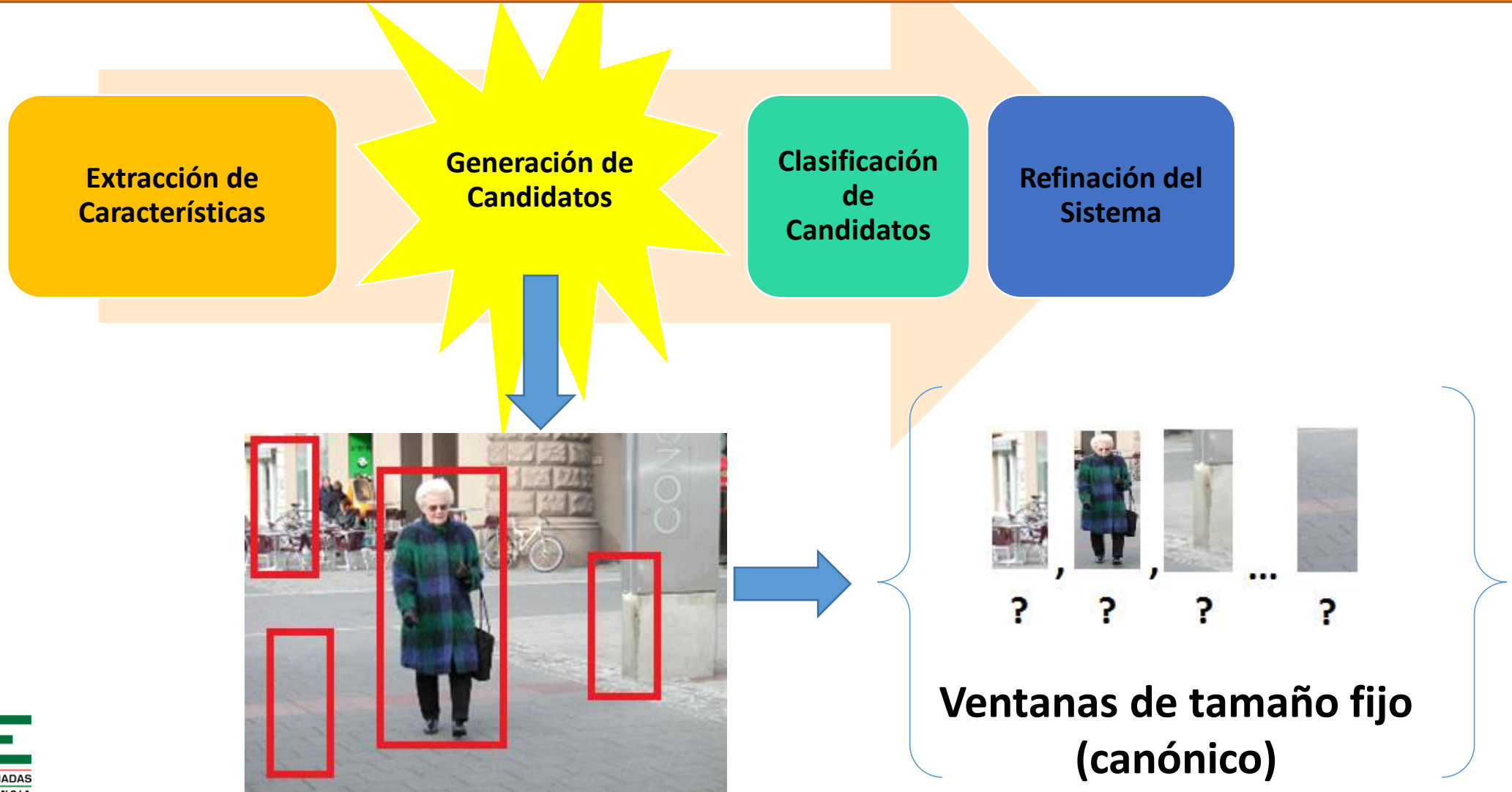
ESPE
UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA

OBTENCIÓN DEL VECTOR DE CARACTERÍSTICAS





METODOLOGÍA





Cómo Generar los Candidatos? (VENTANA DESLIZANTE)

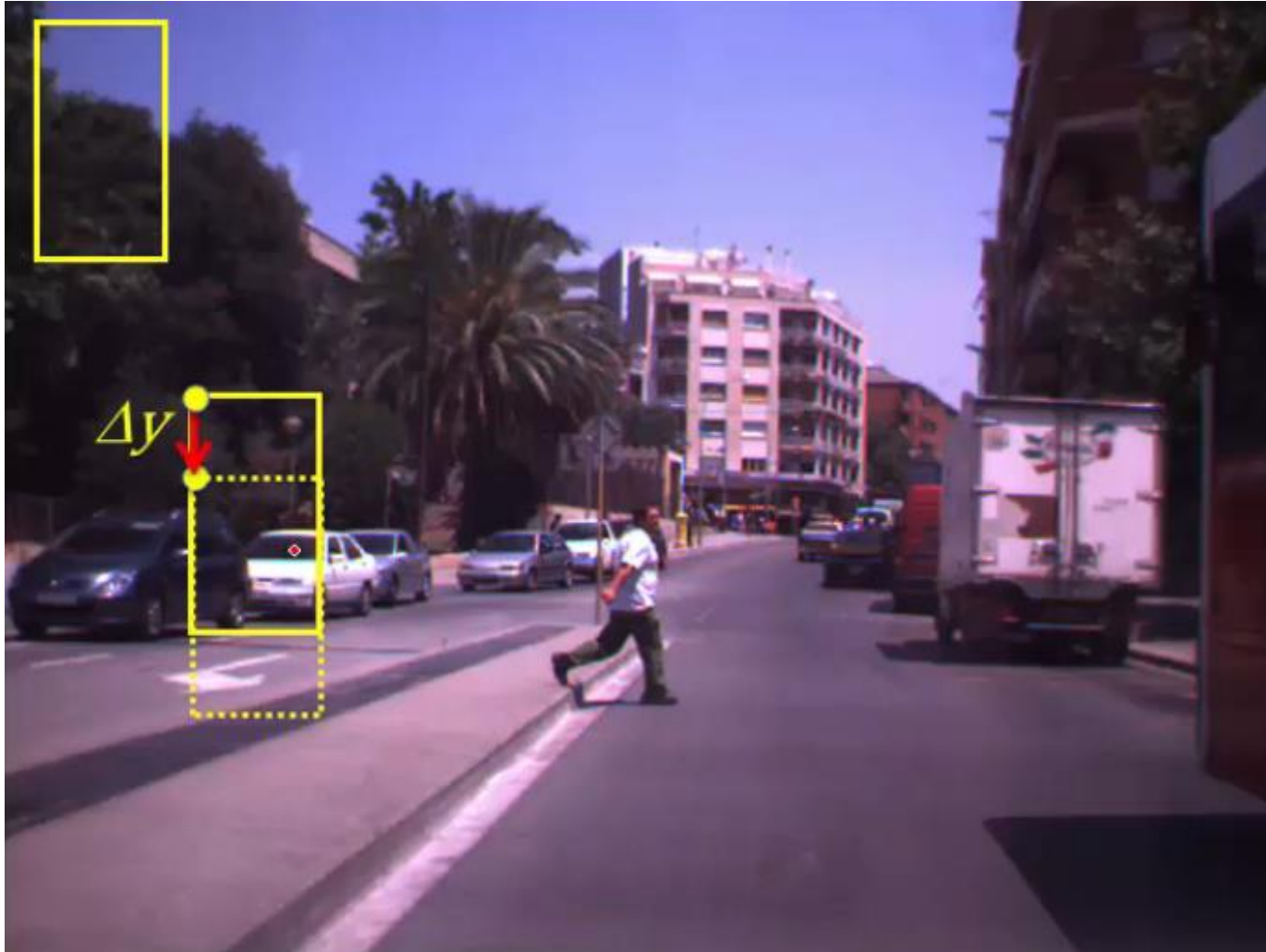


Parámetros:

- Tamaño canónico de la ventana deslizante
- Paso en el eje x(columna): Δx
- Paso en el eje y (fila): Δy



VENTANA DESLIZANTE

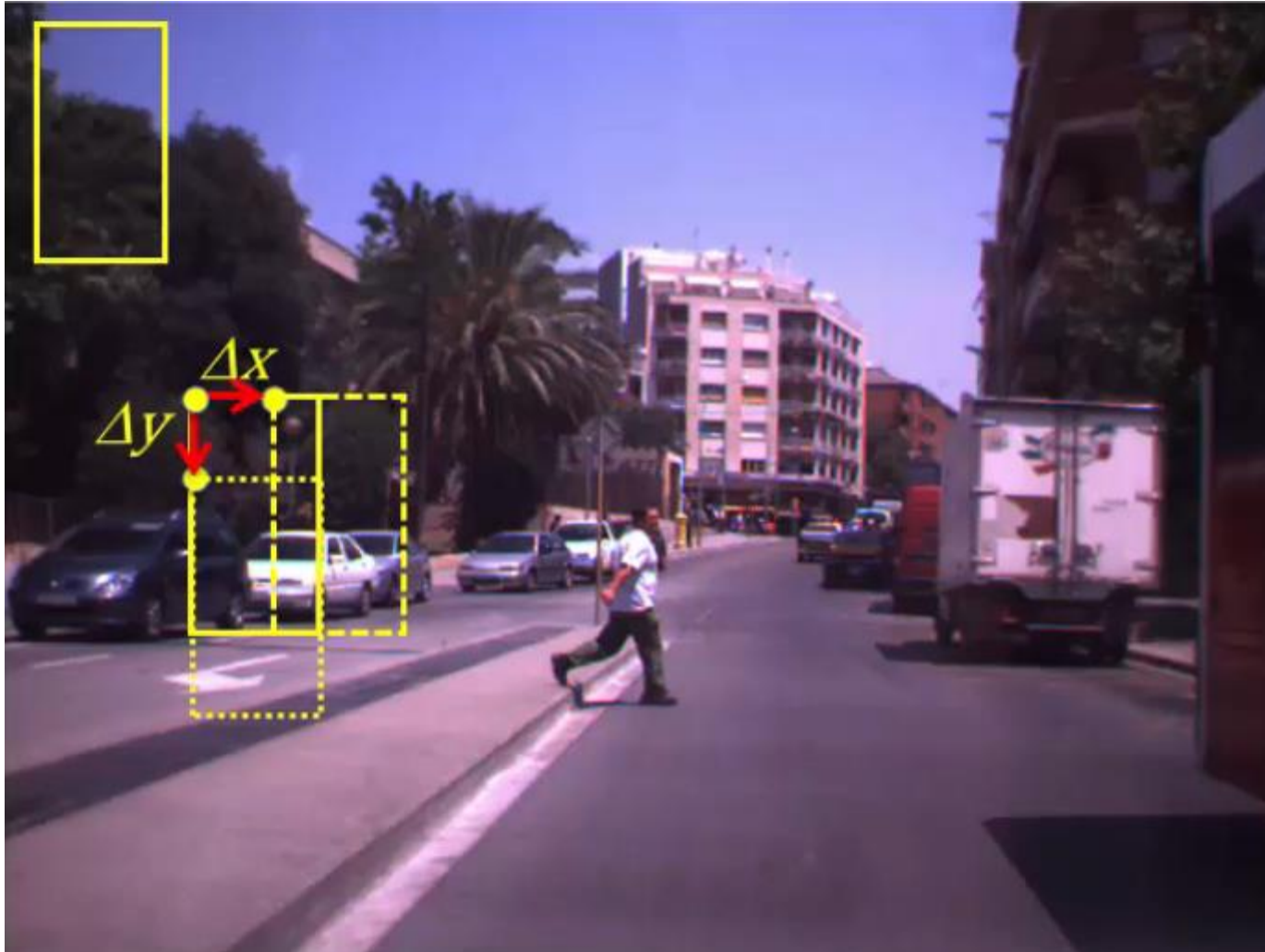


Parámetros:

- Tamaño canónico de la ventana deslizante
- Paso en el eje x(columna): Δx
- Paso en el eje y (fila): Δy



VENTANA DESLIZANTE



Parámetros:

- Tamaño canónico de la ventana deslizante
- Paso en el eje x(columna): Δx
- Paso en el eje y (columna): Δx

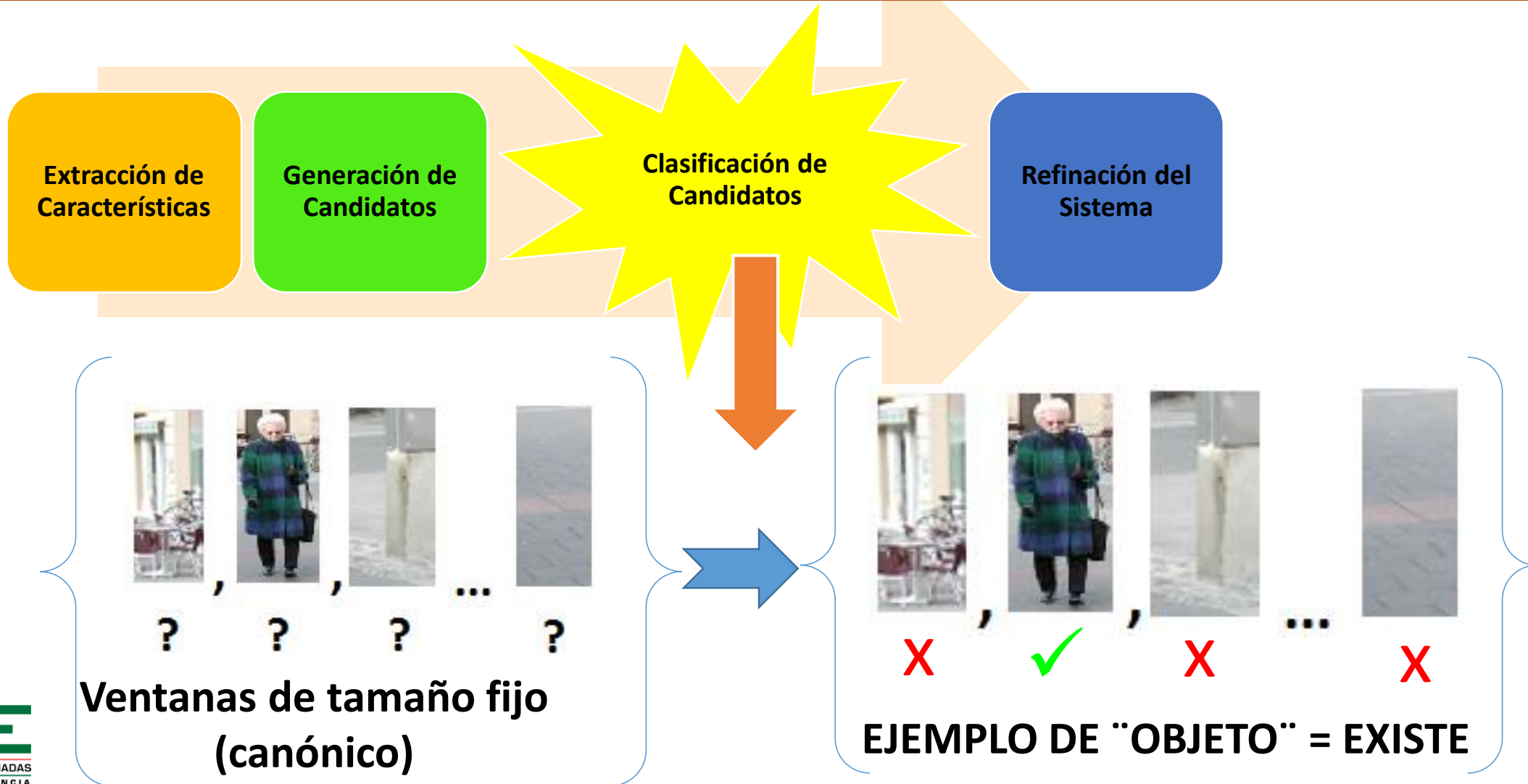


EJEMPLO: VENTANA DESLIZANTE



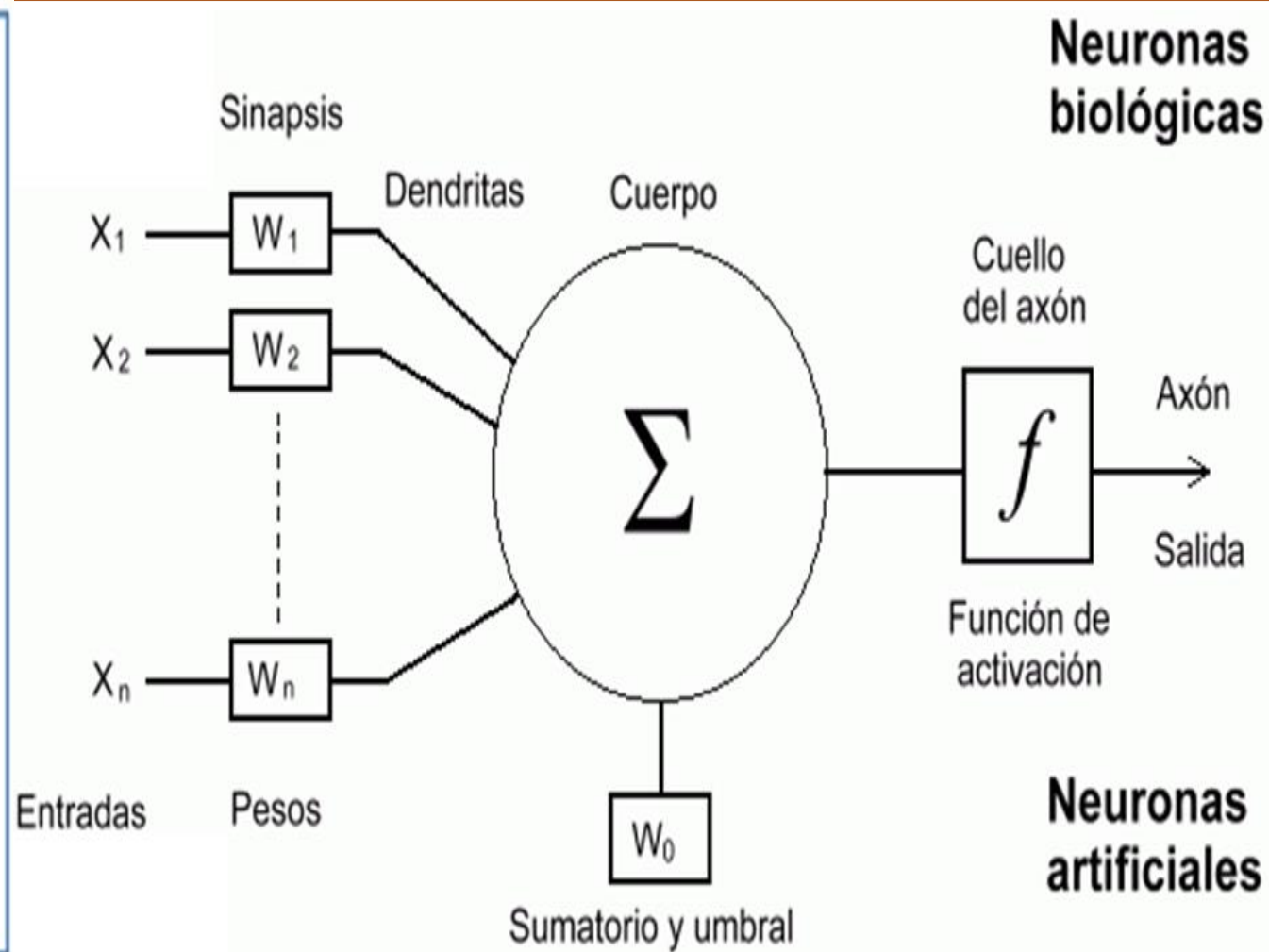
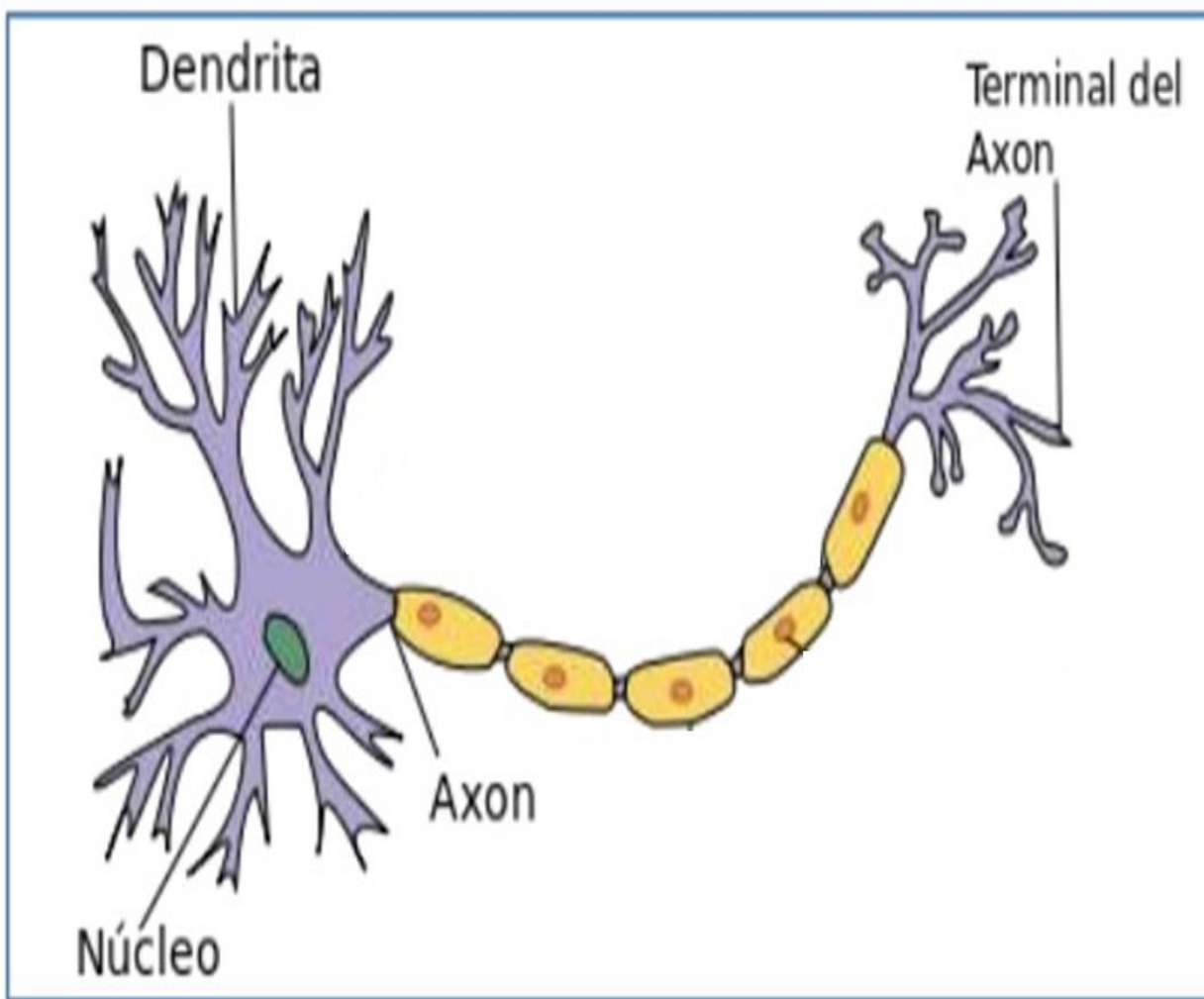


METODOLOGÍA

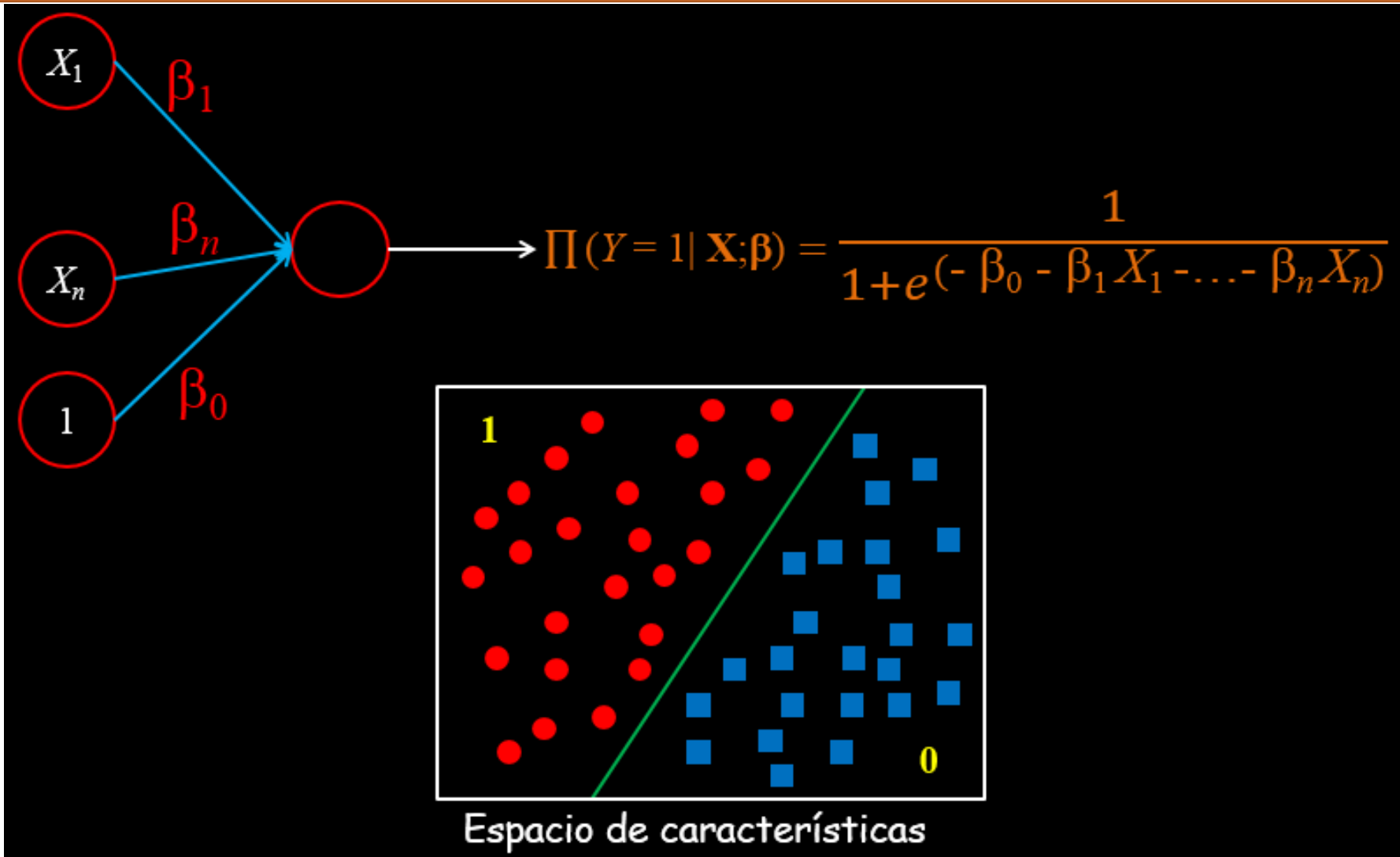




ESTRUCTURA NEURONA BIOLÓGICA vs NEURONA ARTIFICIAL

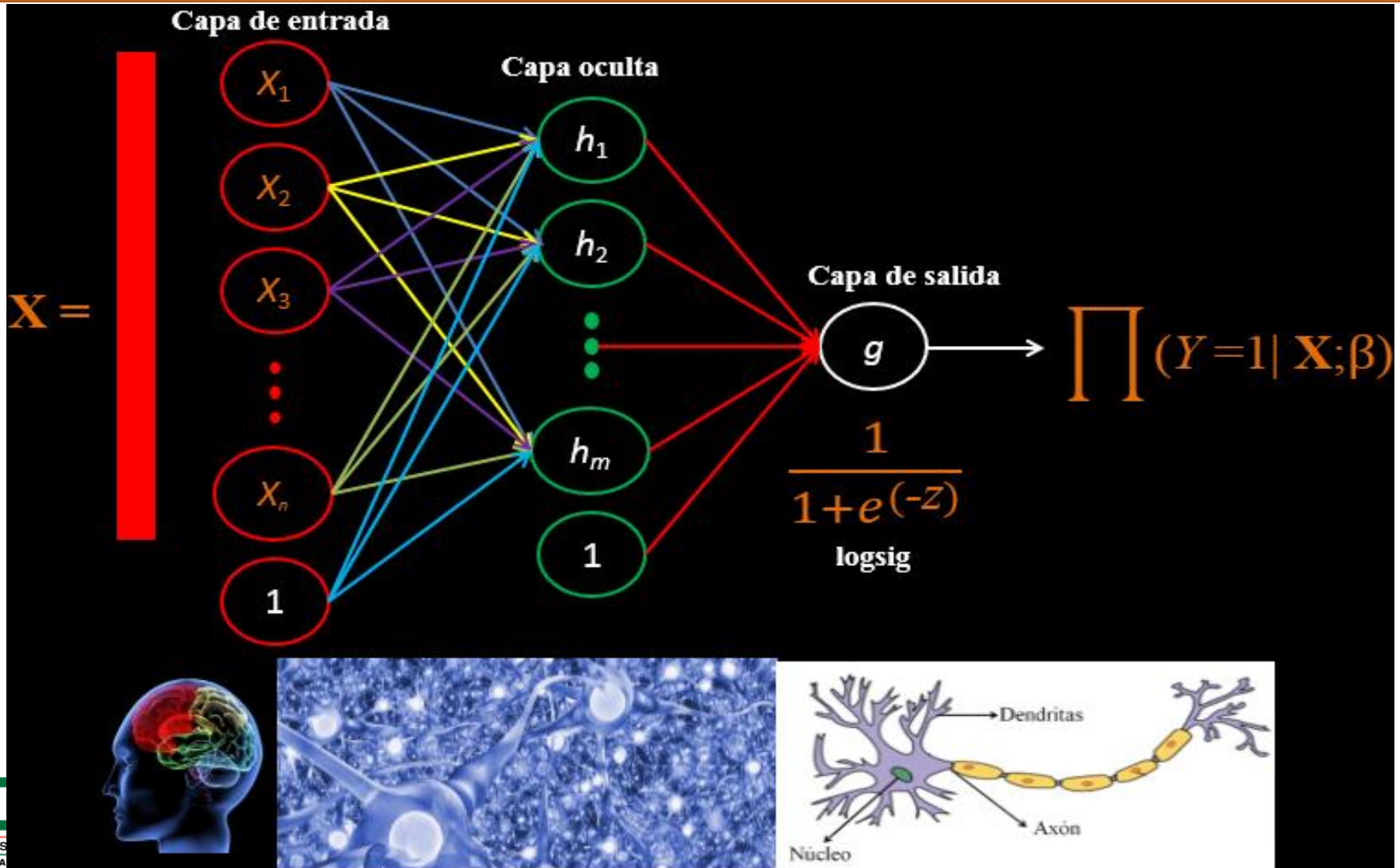


¿QUÉ ES LO QUE HACE CADA NEURONA?





ARQUITECTURA RED NEURONAL ARTIFICIAL





RED NEURONAL (ENTRENAMIENTO)

Vectores de características

Etiquetado

Vector de imagen 1

Vector de imagen 2

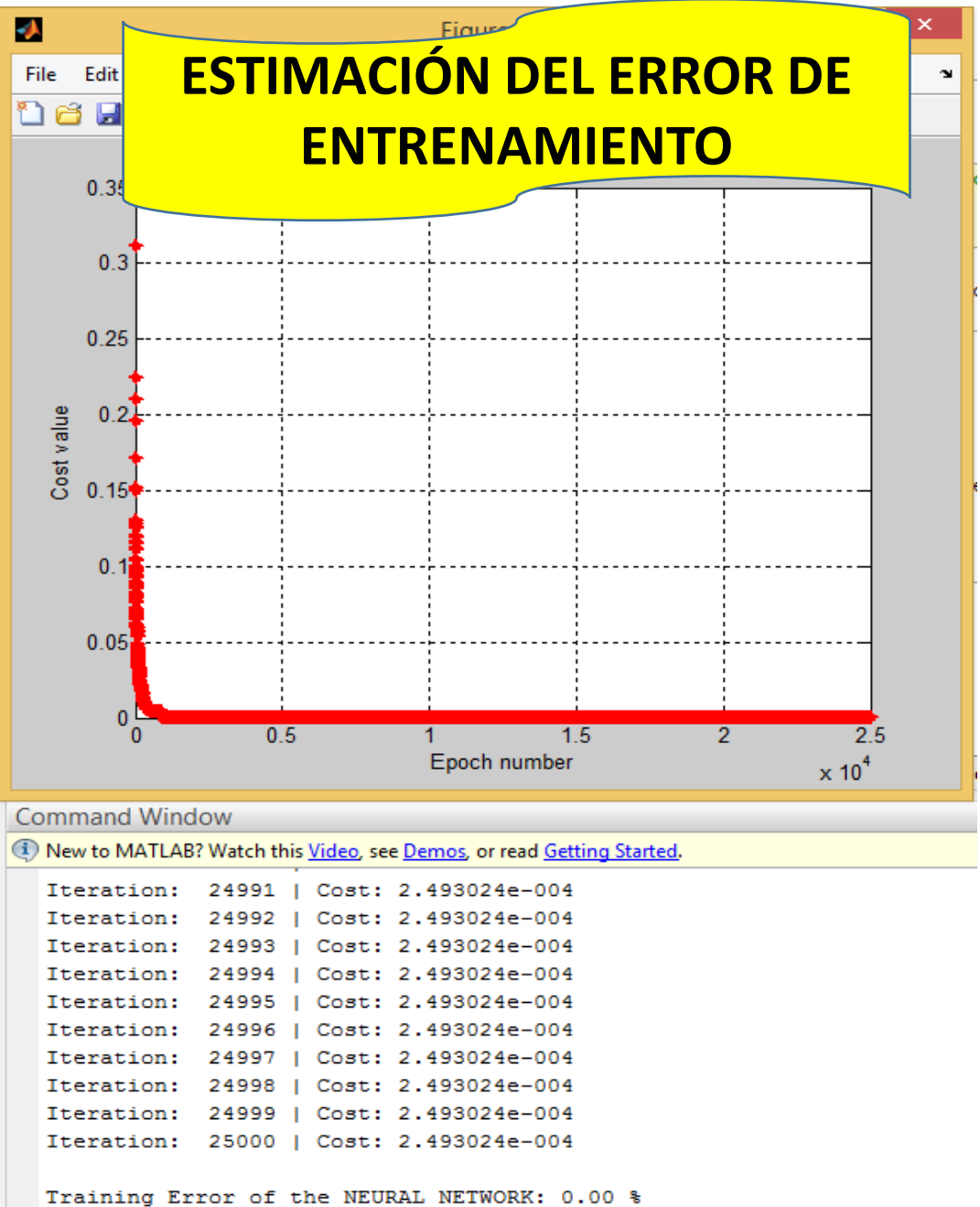
Vector de imagen 2000

$X =$

·
·
·

$Y =$

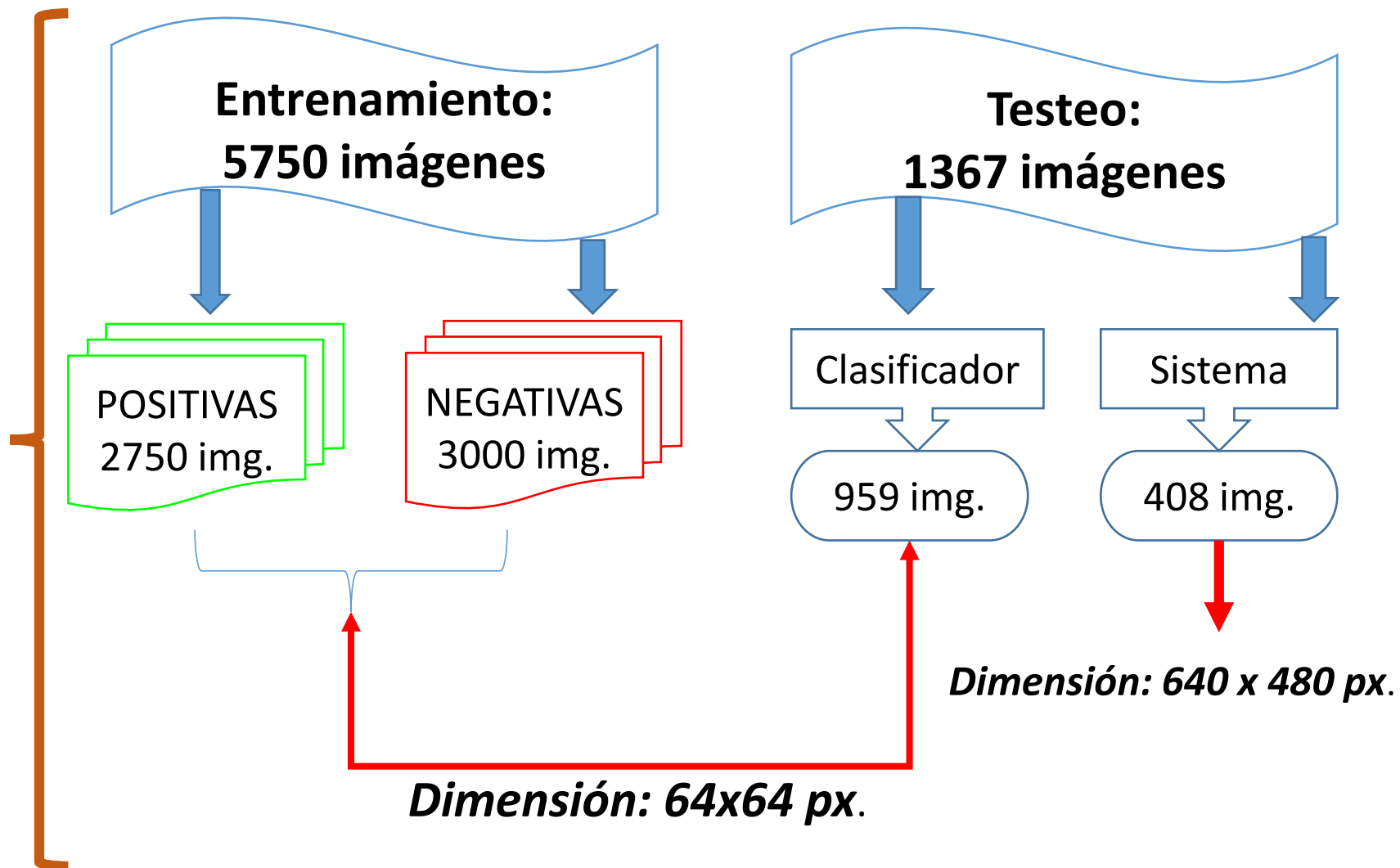
0
0
·
·
·
1



7117 fotografías en total

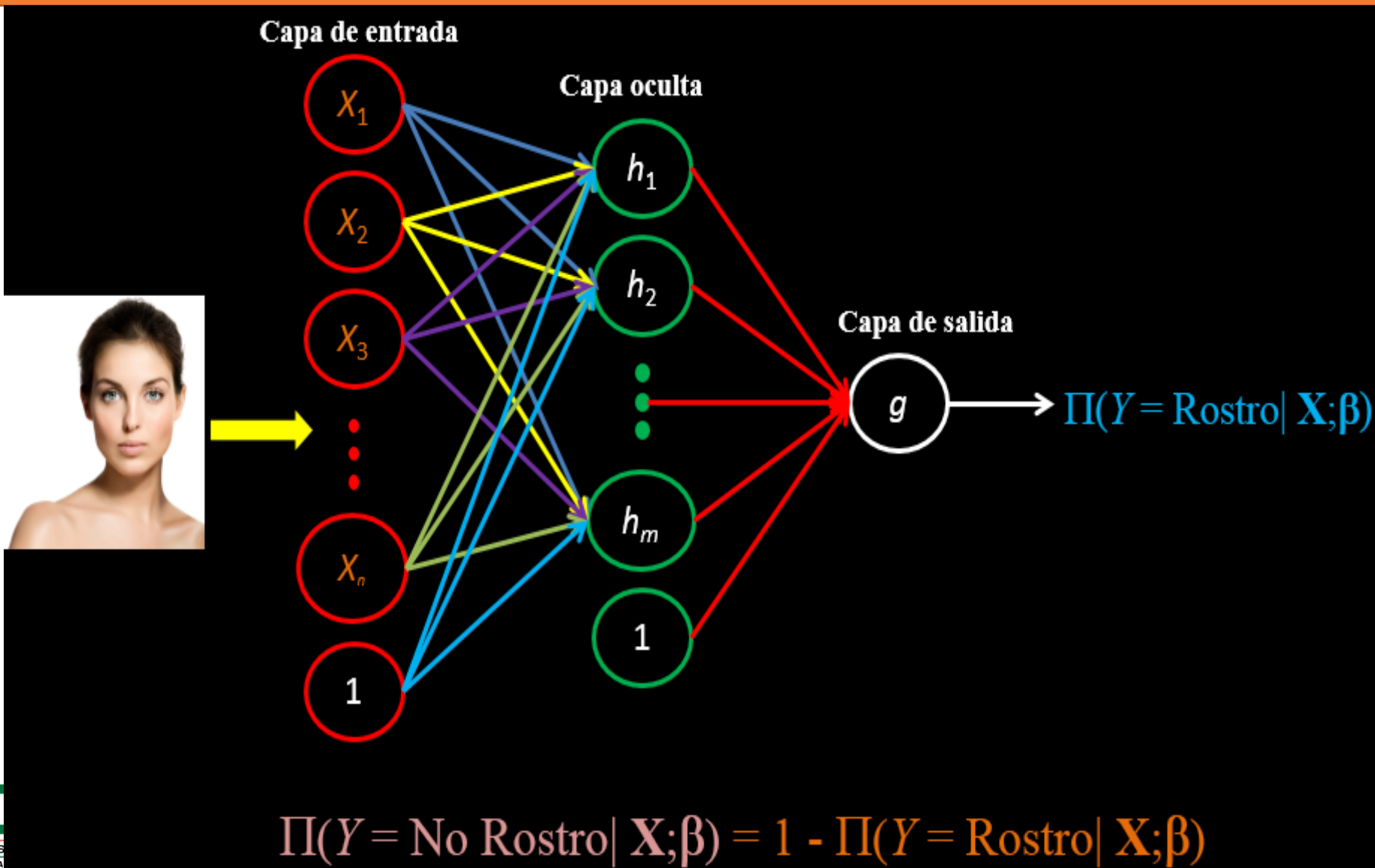


- Imágenes en color.
- Personas entre 12 y 50 años.



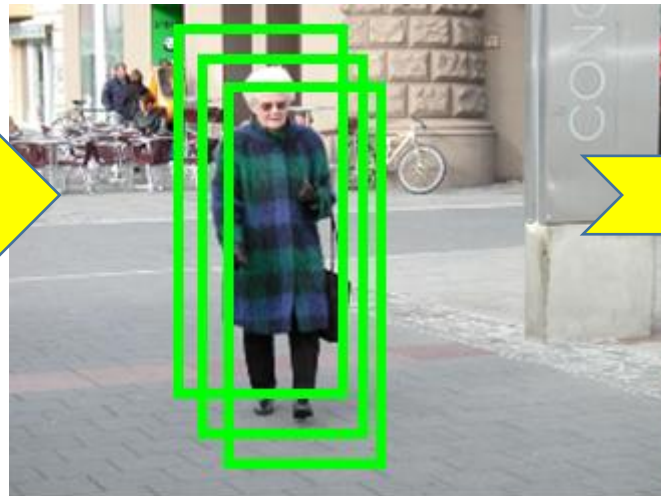
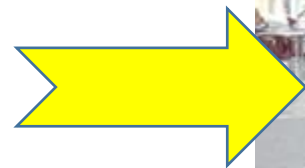
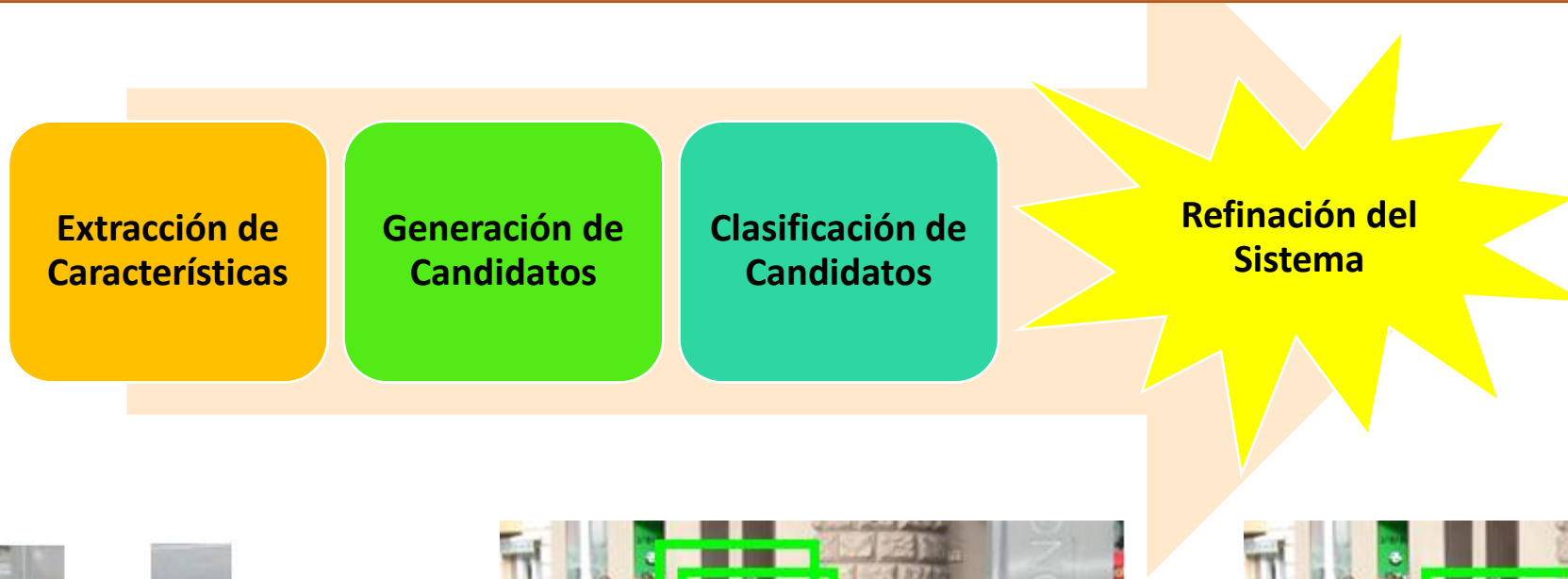


RED NEURONAL





METODOLOGÍA



EJEMPLO DE "OBJETO" = EXISTE

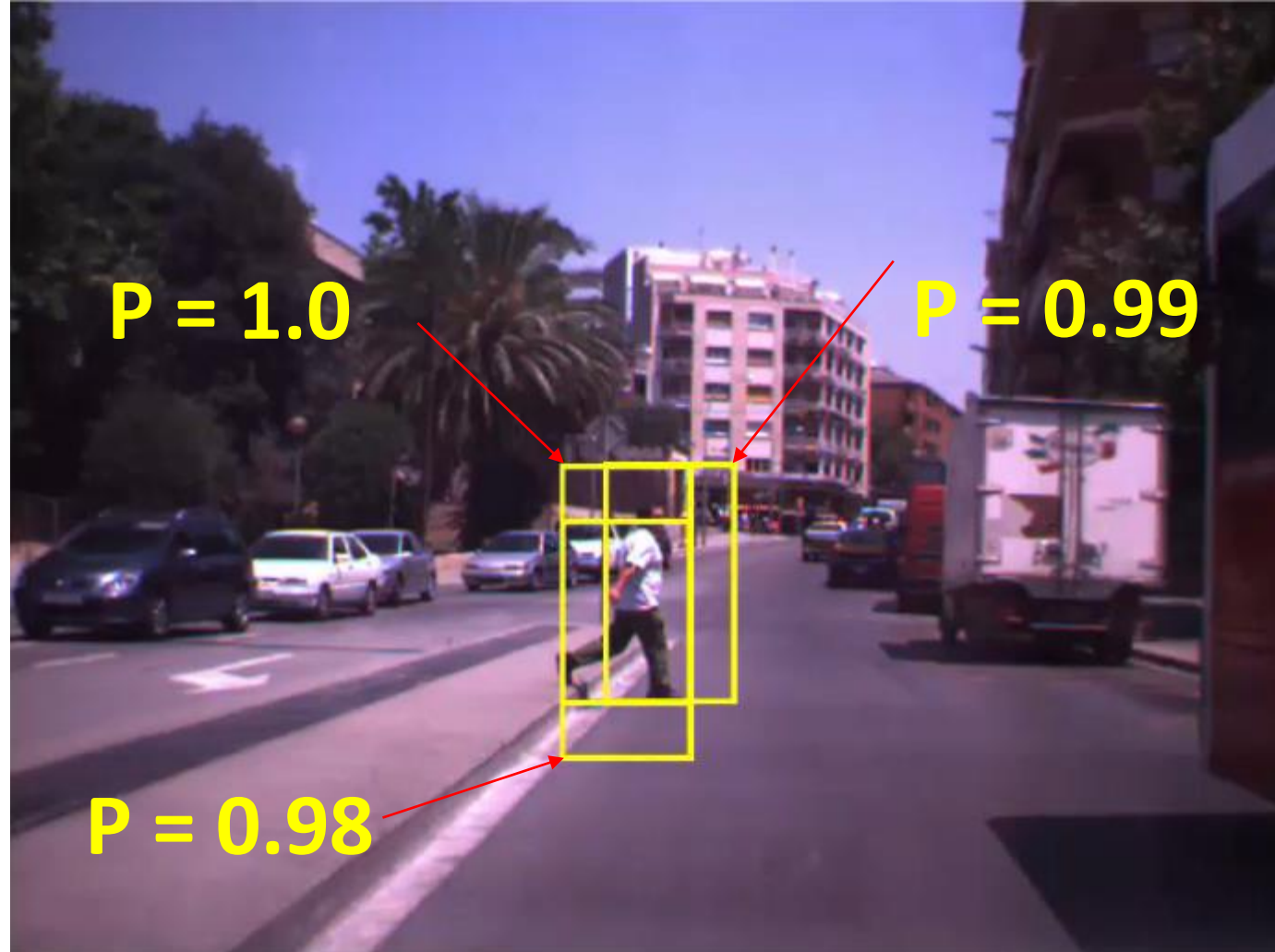


REFINACIÓN DEL SISTEMA





REFINACIÓN DEL SISTEMA





REFINACIÓN DEL SISTEMA





Ejemplo: Refinación – Detección de Rostros

reconocimiento

SUPERPOSICIÓN DE VENTANAS

REFINACIÓN DE VENTANAS REDUNDANTES

Encender cámara Iniciar Apagar



Ejemplo de la Refinación del Sistema con Rostros

reconocimiento

SUPERPOSICIÓN DE VENTANAS

| | |
|------------|--|
| 0.99092843 | |
| 0.99947 | |
| 1.09951 | |
| 0.99989 | |

| | | |
|---------|---------|---------|
| 1 | 0.99996 | |
| 1 | 0.99949 | |
| 1 | 1 | 0.9991 |
| 1 | 1 | 0.99974 |
| 0.99198 | | |
| 0.95 | | |

0.99996
0.9999
0.93536

REFINACIÓN DE VENTANAS REDUNDANTES

| |
|---|
| 1 |
|---|

0.99996

| |
|---|
| 1 |
|---|

Encender cámara

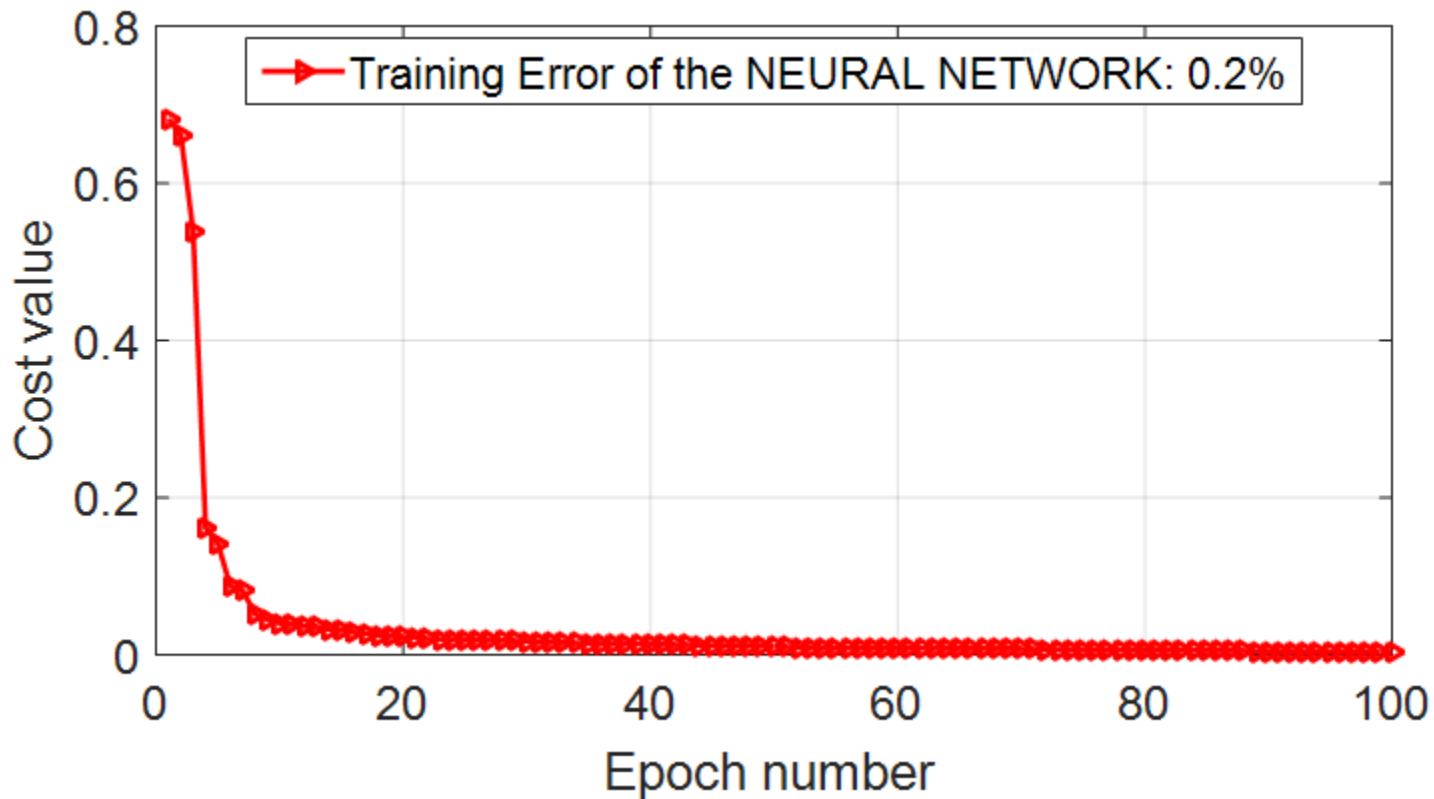
Iniciar

Apagar

RESULTADOS EXPERIMENTALES



Entrenamiento de ANN



Para el entrenamiento usamos 5750 imágenes.

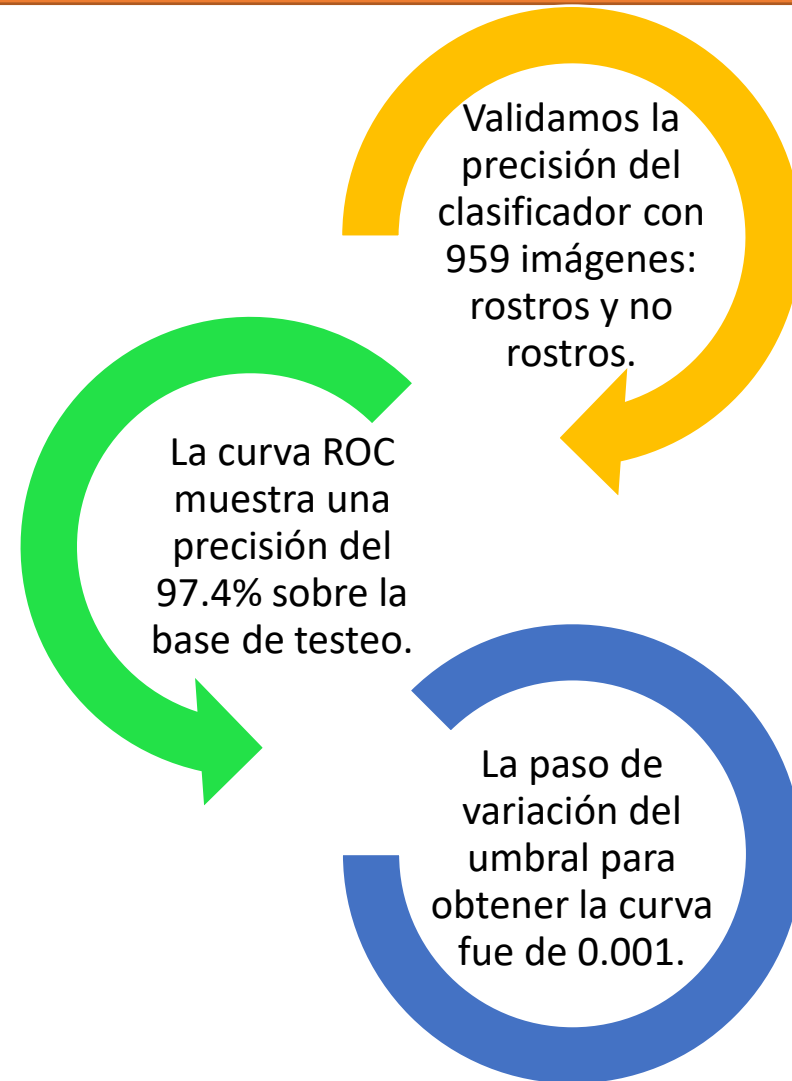
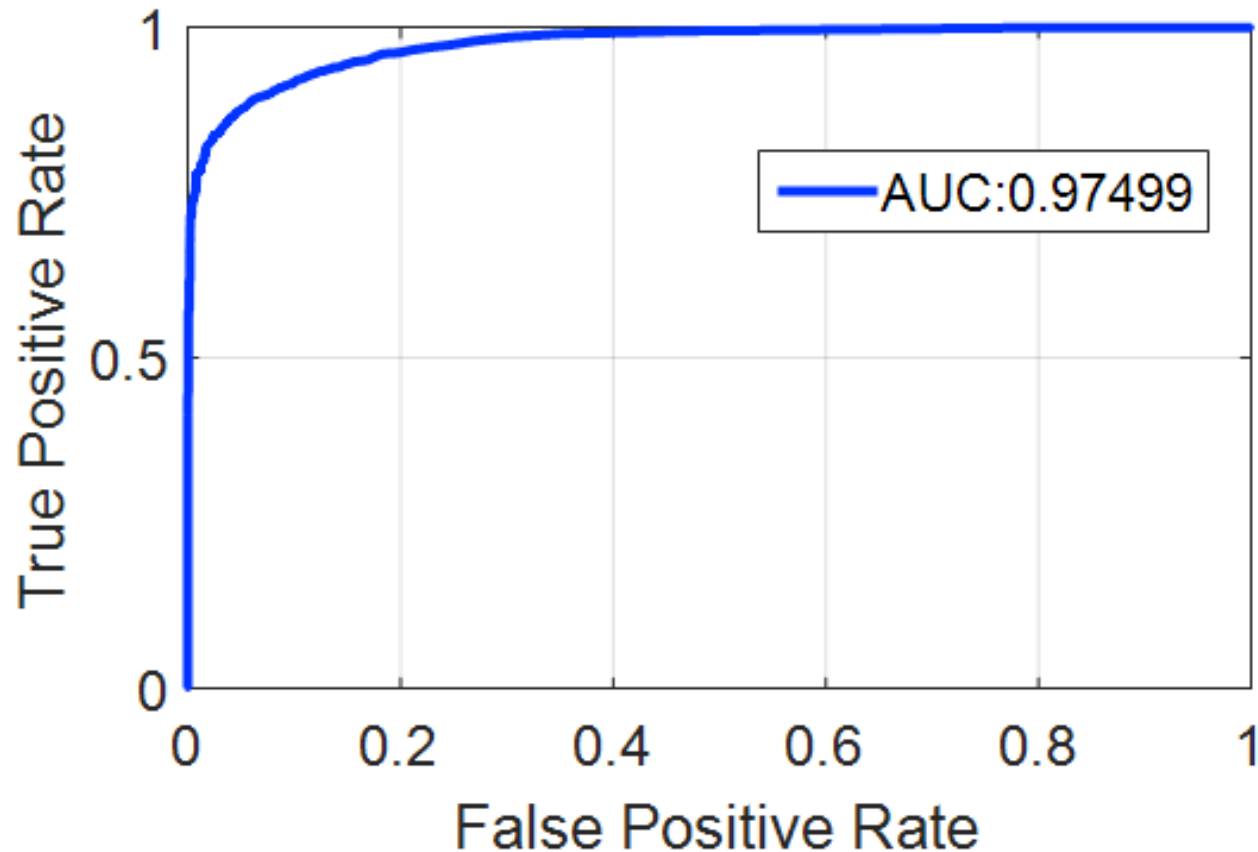
La capa oculta estuvo compuesta de 4 neuronas.

Obtuvimos un error de entrenamiento de 0.2% después de 100 iteraciones.



RESULTADOS EXPERIMENTALES

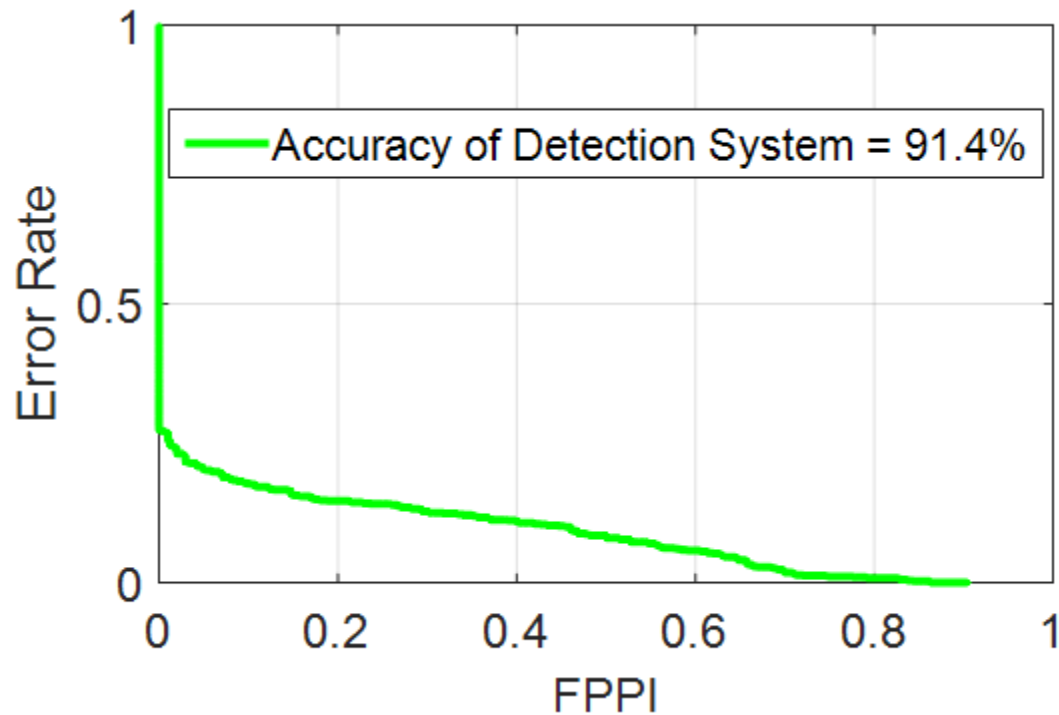
Validación del módulo de clasificación



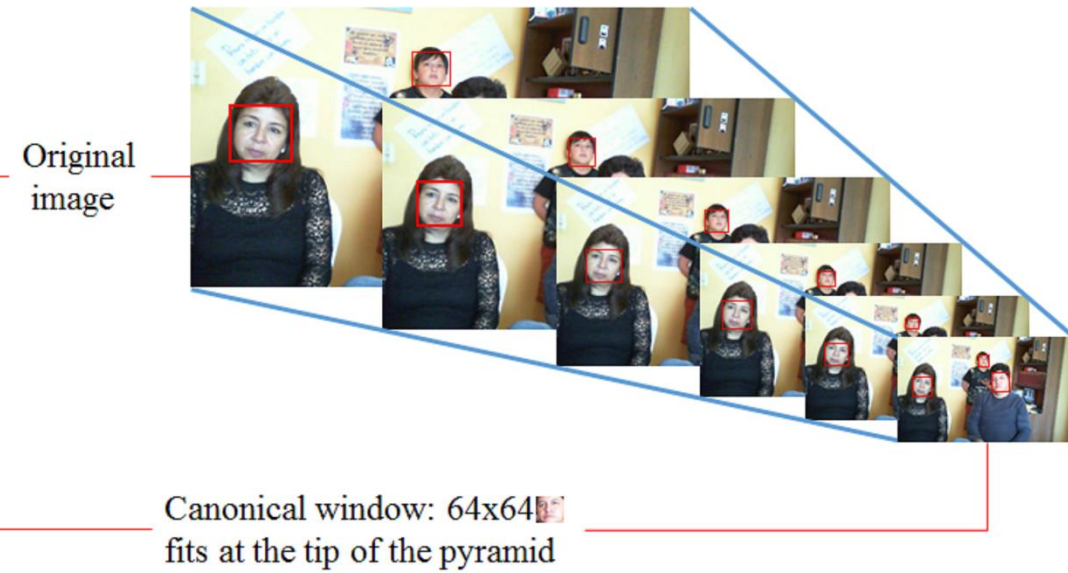


RESULTADOS EXPERIMENTALES

Validación del Sistema Completo



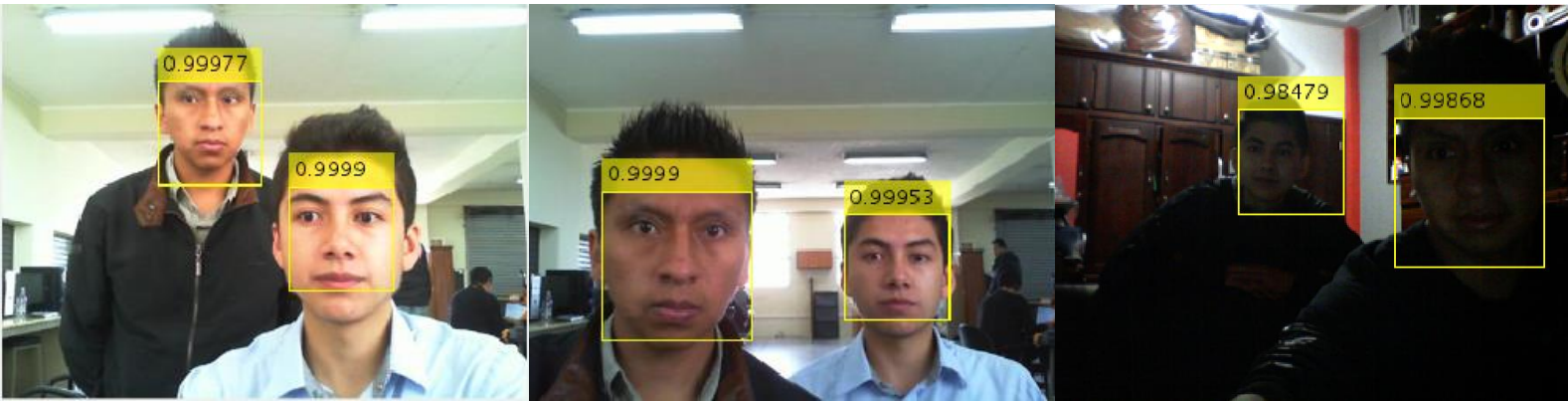
| i | Width | Height |
|---|-------|--------|
| 0 | 320 | 240 |
| 1 | 290 | 218 |
| 2 | 264 | 198 |
| 3 | 240 | 180 |
| 4 | 218 | 163 |
| 5 | 198 | 149 |



RESULTADOS EXPERIMENTALES

Pruebas de detección variando las condiciones luminosas

Tiempo promedio de ejecución de cada modulo que compone el sistema propuesto



(a)

(b)

(c)



(d)

(e)

(f)

| Paso | Tiempo promedio |
|---|-----------------|
| Búsqueda de rostros potenciales en diferentes escalas | 31ms |
| Extracción de características | 16ms |
| Clasificación | 5ms |
| Refinación del sistema | 29ms |
| Total | 81ms |



CONCLUSIONES

- En este trabajo, presentamos un sistema de detección de objetos en tiempo real. Usamos un extractor de características basado en histogramas de gradientes orientados.
- Probamos este sistema para detectar rostros, mostrando una tasa de detección del 91.4%. Aunque utilizamos una red neuronal superficial con solo 4 neuronas en la capa oculta, obtuvimos un alto rendimiento en entornos no estructurados que incluían variaciones en el luz, postura y oclusión.
- Probamos el modelo propuesto utilizando recursos computacionales no sofisticados. El tiempo promedio de procesamiento del algoritmo completo es de 81ms para cada cuadro de video de 320x240 píxeles. Para lograr esta velocidad, ejecutamos las diferentes escalas de detección en paralelo, combinando lenguajes de alto y bajo nivel (MATLAB y C ++).
- También ponemos a disposición del público los juegos de entrenamiento y prueba que utilizamos para este trabajo en <http://electronica-el.espe.edu.ec/actividad-estudiantil/face-database/>.
- El trabajo futuro incluye probar otros clasificadores diferentes de las redes neuronales.