

Semáforo Inteligente Utilizando Procesamiento Digital de Imágenes Controlado por Medio del Perceptrón de Rosenblatt

Mario Rosas O. Natalia Sánchez P. Dr. David Tinoco V.

FESC, UNAM

Mayo, 2021



① Introducción

② Marco teórico

③ Descripción del sistema

④ Experimentación

⑤ Resultados

⑥ Conclusiones

① Introducción

Control inteligente en semáforos de paso

Semáforos tradicionales vs. semáforos inteligentes

Trabajos relacionados

② Marco teórico

③ Descripción del sistema

④ Experimentación

⑤ Resultados

⑥ Conclusiones

① Introducción

Control inteligente en semáforos de paso

Semáforos tradicionales vs. semáforos inteligentes

Trabajos relacionados

② Marco teórico

③ Descripción del sistema

④ Experimentación

⑤ Resultados

⑥ Conclusiones

Control inteligente en semáforos de paso

- Sistemas de ingeniería para control automático: IA, BD, IoT, I4.0.

Control inteligente en semáforos de paso

- Sistemas de ingeniería para control automático: IA, BD, IoT, I4.0.
- Elementos de una ciudad inteligente.
- Semáforo para el control del flujo vehicular:

Control inteligente en semáforos de paso

- Sistemas de ingeniería para control automático: IA, BD, IoT, I4.0.
- Elementos de una ciudad inteligente.
- Semáforo para el control del flujo vehicular:
 - Reacción inteligente

Control inteligente en semáforos de paso

- Sistemas de ingeniería para control automático: IA, BD, IoT, I4.0.
- Elementos de una ciudad inteligente.
- Semáforo para el control del flujo vehicular:
 - Reacción inteligente
 - Adaptativo en tiempo real

Control inteligente en semáforos de paso

- Sistemas de ingeniería para control automático: IA,BD. IoT, I4.0.
- Elementos de una ciudad inteligente.
- Semáforo para el control del flujo vehicular:
 - Reacción inteligente
 - Adaptativo en tiempo real
- Interacción segura entre peatones y automóviles.

Control inteligente en semáforos de paso

- Sistemas de ingeniería para control automático: IA, BD, IoT, I4.0.
- Elementos de una ciudad inteligente.
- Semáforo para el control del flujo vehicular:
 - Reacción inteligente
 - Adaptativo en tiempo real
- Interacción segura entre peatones y automóviles.
- Optimizar tiempos de paso

Control inteligente en semáforos de paso

- Sistemas de ingeniería para control automático: IA, BD, IoT, I4.0.
- Elementos de una ciudad inteligente.
- Semáforo para el control del flujo vehicular:
 - Reacción inteligente
 - Adaptativo en tiempo real
- Interacción segura entre peatones y automóviles.
- Optimizar tiempos de paso
- Mejorar la eficiencia de la movilidad en la ciudad

Control inteligente en semáforos de paso

- Sistemas de ingeniería para control automático: IA, BD, IoT, I4.0.
- Elementos de una ciudad inteligente.
- Semáforo para el control del flujo vehicular:
 - Reacción inteligente
 - Adaptativo en tiempo real
- Interacción segura entre peatones y automóviles.
- Optimizar tiempos de paso
- Mejorar la eficiencia de la movilidad en la ciudad
- Sistema propuesto:

Control inteligente en semáforos de paso

- Sistemas de ingeniería para control automático: IA, BD, IoT, I4.0.
- Elementos de una ciudad inteligente.
- Semáforo para el control del flujo vehicular:
 - Reacción inteligente
 - Adaptativo en tiempo real
- Interacción segura entre peatones y automóviles.
- Optimizar tiempos de paso
- Mejorar la eficiencia de la movilidad en la ciudad
- Sistema propuesto:
 - Sistema para cambio de luz de semáforo con imágenes en tiempo real, con pre-procesamiento y toma de decisión utilizando un perceptrón.

① Introducción

Control inteligente en semáforos de paso

Semáforos tradicionales vs. semáforos inteligentes

Trabajos relacionados

② Marco teórico

③ Descripción del sistema

④ Experimentación

⑤ Resultados

⑥ Conclusiones

Semáforos tradicionales vs. semáforos inteligentes

Características de los semáforos

Tradicionales

- Tiempo de activación y desactivación
- No consideran las condiciones de flujo
- Controladores simples

Inteligentes

- Responde a las necesidades del entorno
- Tratan dinámicamente las situaciones
- Sensores y análisis inteligente

Cuadro 1: Comparativa entre tipos de semáforos

① Introducción

Control inteligente en semáforos de paso

Semáforos tradicionales vs. semáforos inteligentes

Trabajos relacionados

② Marco teórico

③ Descripción del sistema

④ Experimentación

⑤ Resultados

⑥ Conclusiones

Trabajos relacionados

Cuadro 2: Propuestas de semáforos inteligentes existentes.

Algoritmos y técnicas para SCIs	
Control Vehicular	Control Peatonal
<ul style="list-style-type: none">■ Aprendizaje por Refuerzo■ Sistemas Expertos■ Aprendizaje Profundo■ Control Difuso■ ENN <ul style="list-style-type: none">● La propuesta presentada: un procesamiento que requiera bajo poder computacional y funcional haya o no conexión a internet.	<ul style="list-style-type: none">■ Algoritmos Genéticos■ LSTM■ Q multi-agente distribuido■ Redes Neuronales Convolucionales

① Introducción

② Marco teórico

Algoritmo

③ Descripción del sistema

④ Experimentación

⑤ Resultados

⑥ Conclusiones

① Introducción

② Marco teórico Algoritmo

③ Descripción del sistema

④ Experimentación

⑤ Resultados

⑥ Conclusiones

Algoritmo

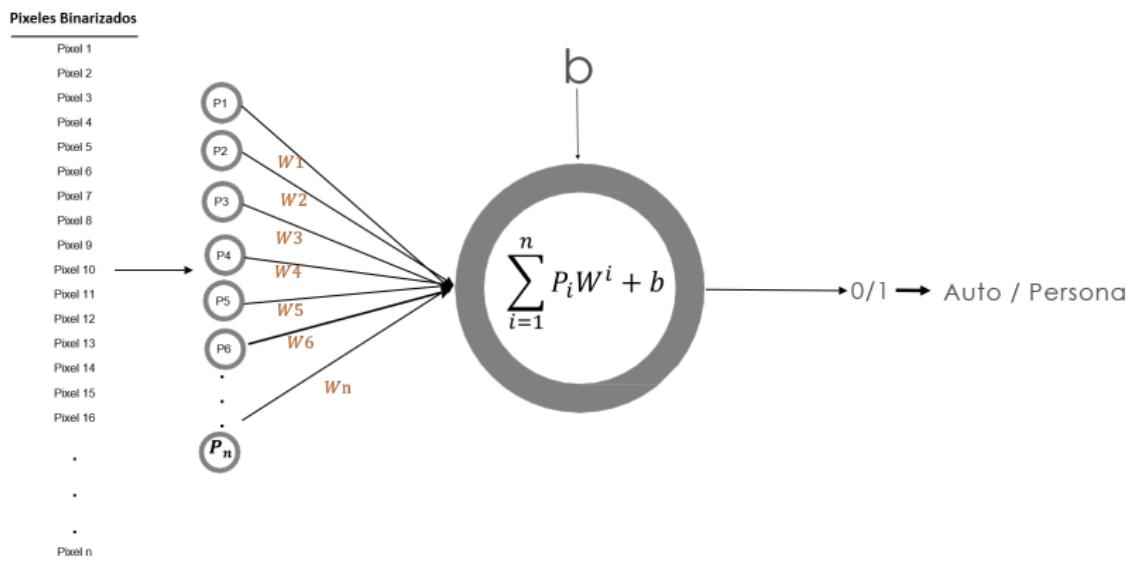


Figura 1: Perceptrón de Rosenblatt (1958)

① Introducción

② Marco teórico

③ Descripción del sistema

Descripción general del sistema

Adquisición y almacenamiento de datos

④ Experimentación

⑤ Resultados

⑥ Conclusiones

① Introducción

② Marco teórico

③ Descripción del sistema

Descripción general del sistema

Adquisición y almacenamiento de datos

④ Experimentación

⑤ Resultados

⑥ Conclusiones

Funcionamiento general del sistema

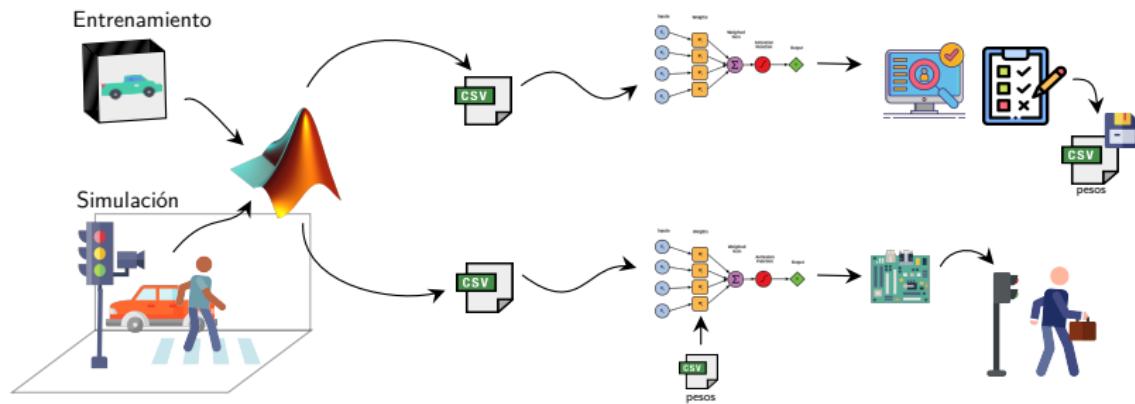


Figura 2: Diagrama del sistema

① Introducción

② Marco teórico

③ Descripción del sistema

Descripción general del sistema

Adquisición y almacenamiento de datos

④ Experimentación

⑤ Resultados

⑥ Conclusiones

Imágenes de ejemplo



Figura 3: Representación de coches y personas.

Tablas de Datos

Cuadro 3: Tablas de imágenes utilizadas.

Imágenes Utilizadas Aproximación 1					
	Experimento 1			Experimento 2	
	Entrenamiento	Validación	Pruebas en tiempo real	Entrenamiento	Validación
Autos	24	12	10	50	15
Personas	24	12	10	50	15

Imágenes Utilizadas Aproximación 2				
	Experimento 1			
	Entrenamiento	Validación	Pruebas en tiempo real	Fuera de distribución
Autos	130	45	30	7
Personas	130	45	30	8

Aproximación 1 - Experimento 1

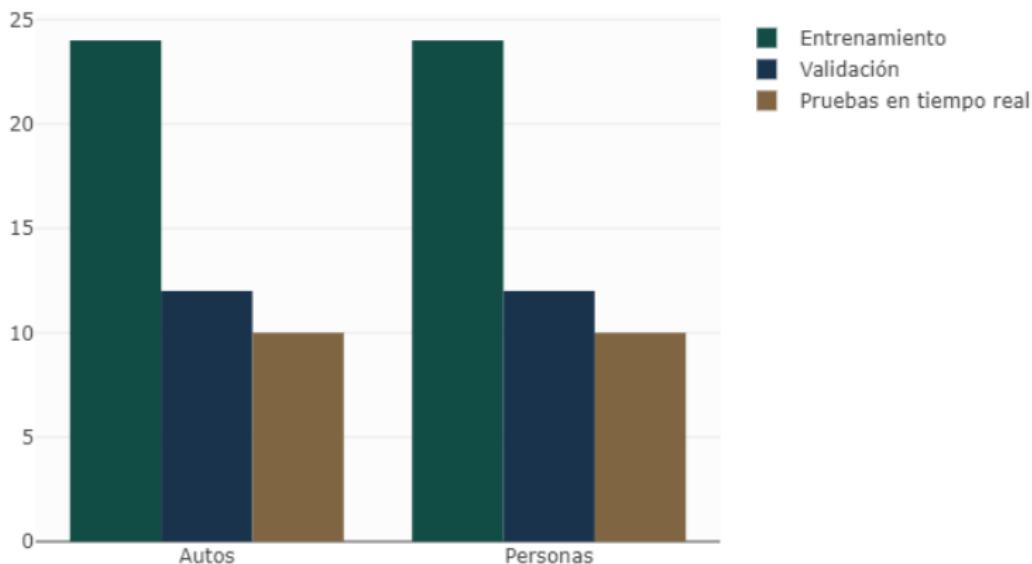


Figura 4: Distribución de imágenes en el conjunto de datos E1

Aproximación 1 - Experimento 2

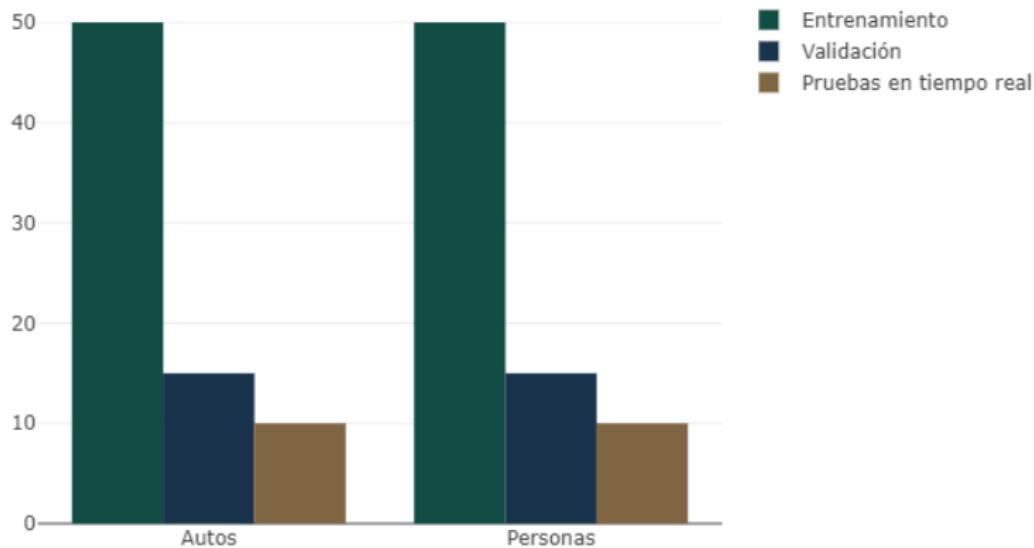


Figura 5: Distribución de imágenes en el conjunto de datos E2

Aproximación 2

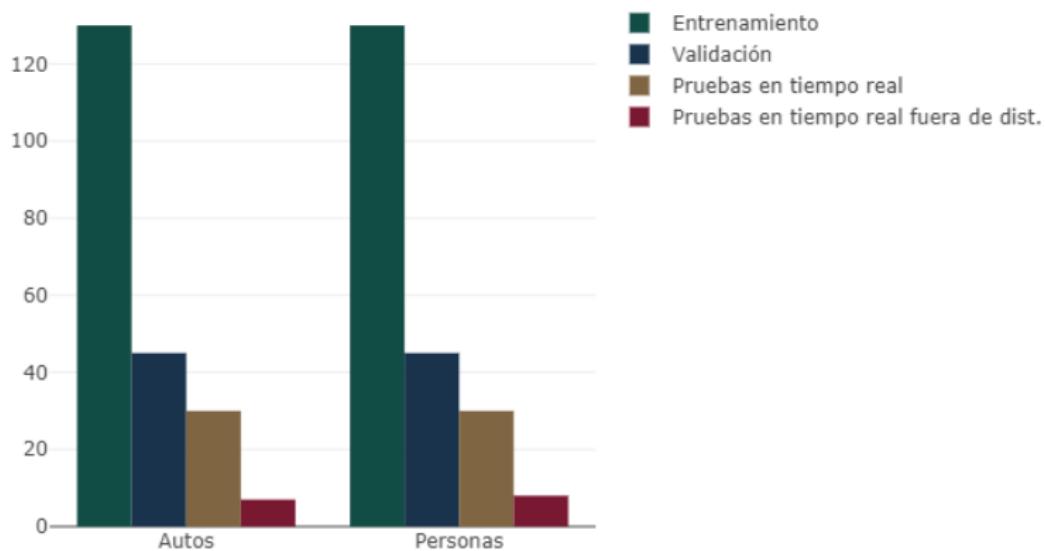


Figura 6: Distribución de imágenes en el conjunto de datos A2

Método de extracción de características

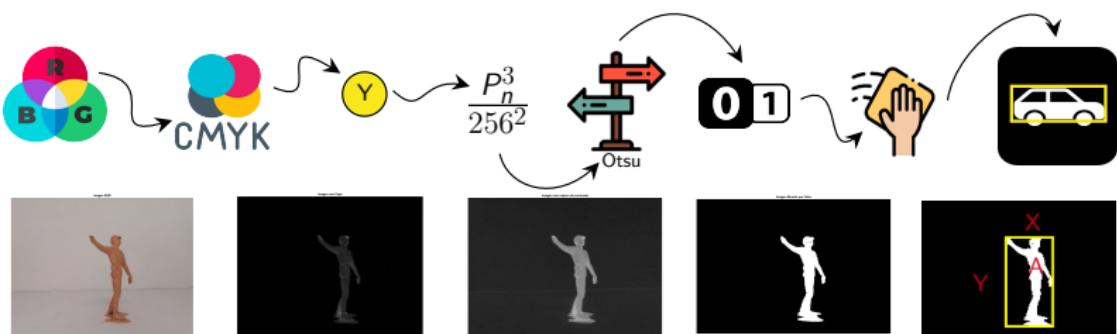


Figura 7: Diagrama del algoritmo

Entrenamiento y validación del perceptrón

- Convertir cada imagen binarizada de matriz a vector.

Entrenamiento y validación del perceptrón

- Convertir cada imagen binarizada de matriz a vector.
- Se forma una matriz con todas las imágenes de entrenamiento (cada fila es una imagen binarizada).

Entrenamiento y validación del perceptrón

- Convertir cada imagen binarizada de matriz a vector.
- Se forma una matriz con todas las imágenes de entrenamiento (cada fila es una imagen binarizada).
- Se construye el vector de salida conforme se crea la matriz de imágenes.

Entrenamiento y validación del perceptrón

- Convertir cada imagen binarizada de matriz a vector.
- Se forma una matriz con todas las imágenes de entrenamiento (cada fila es una imagen binarizada).
- Se construye el vector de salida conforme se crea la matriz de imágenes.
- Se realiza el entrenamiento del perceptrón utilizando la matriz y el vector de salidas.

Entrenamiento y validación del perceptrón

- Convertir cada imagen binarizada de matriz a vector.
- Se forma una matriz con todas las imágenes de entrenamiento (cada fila es una imagen binarizada).
- Se construye el vector de salida conforme se crea la matriz de imágenes.
- Se realiza el entrenamiento del perceptrón utilizando la matriz y el vector de salidas.
- Cuando se alcanza la convergencia de acuerdo al esquema de entrenamiento se guardan los pesos y el sesgo.

Entrenamiento y validación del perceptrón

- Convertir cada imagen binarizada de matriz a vector.
- Se forma una matriz con todas las imágenes de entrenamiento (cada fila es una imagen binarizada).
- Se construye el vector de salida conforme se crea la matriz de imágenes.
- Se realiza el entrenamiento del perceptrón utilizando la matriz y el vector de salidas.
- Cuando se alcanza la convergencia de acuerdo al esquema de entrenamiento se guardan los pesos y el sesgo.
- Con el conjunto de validación se mide la capacidad de generalización del sistema.

Entrenamiento y validación del perceptrón

- Convertir cada imagen binarizada de matriz a vector.
- Se forma una matriz con todas las imágenes de entrenamiento (cada fila es una imagen binarizada).
- Se construye el vector de salida conforme se crea la matriz de imágenes.
- Se realiza el entrenamiento del perceptrón utilizando la matriz y el vector de salidas.
- Cuando se alcanza la convergencia de acuerdo al esquema de entrenamiento se guardan los pesos y el sesgo.
- Con el conjunto de validación se mide la capacidad de generalización del sistema.
- Con los pesos y sesgo obtenidos ya es posible realizar la simulación del sistema en tiempo real.

Entrenamiento y validación de la red

- Extraer las características de cada imagen binarizada.
- Se forma una matriz con las características extraídas de la imágenes de entrenamiento (3 características por imagen).
- Se construye el vector de salida conforme se forma la matriz de características.
- Se realiza el entrenamiento del perceptrón utilizando la matriz y el vector de salidas.
- Cuando se alcanza la convergencia de acuerdo al esquema de entrenamiento se guardan los pesos y el sesgo.
- Con el conjunto de validación se mide la capacidad de generalización del sistema.
- Con los pesos y sesgo obtenidos ya es posible realizar la simulación del sistema en tiempo real.

Simulación (Pruebas en tiempo real)

- Se obtiene la imagen en tiempo real.

Simulación (Pruebas en tiempo real)

- Se obtiene la imagen en tiempo real.
- Se binariza la imagen y se obtienen sus características.

Simulación (Pruebas en tiempo real)

- Se obtiene la imagen en tiempo real.
- Se binariza la imagen y se obtienen sus características.
- Se procesan las características por medio del perceptrón el cual emite una respuesta.

Simulación (Pruebas en tiempo real)

- Se obtiene la imagen en tiempo real.
- Se binariza la imagen y se obtienen sus características.
- Se procesan las características por medio del perceptrón el cual emite una respuesta.
- Esta respuesta es utilizada para tomar la decisión del cambio de luces del semáforo, implementado mediante LEDs y un microcontrolador.

① Introducción

② Marco teórico

③ Descripción del sistema

④ Experimentación

Aproximación 1

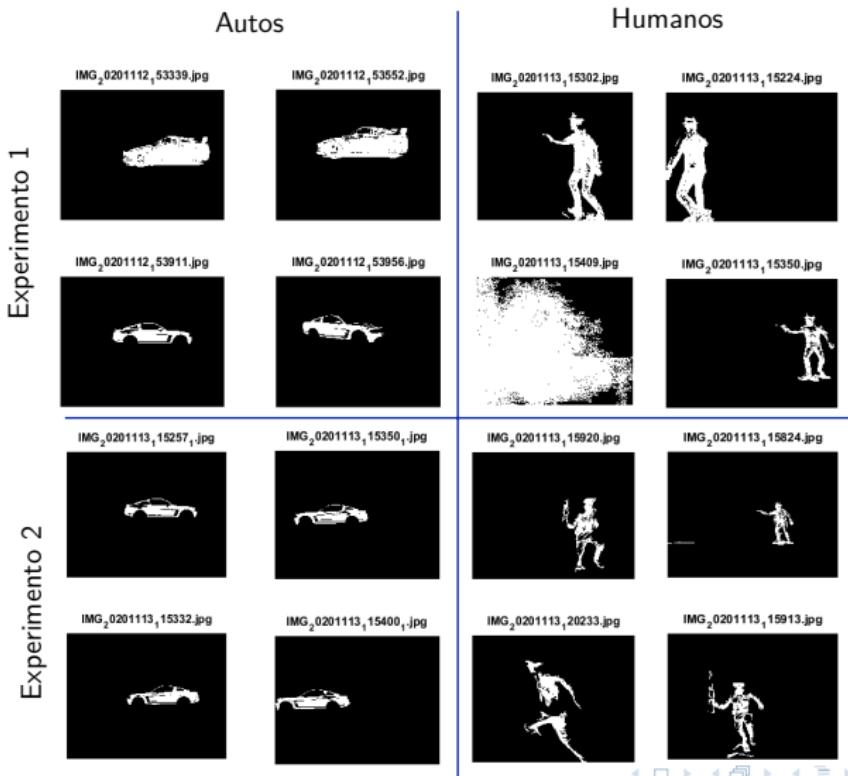
Aproximación 2

Experimentación en tiempo real

⑤ Resultados

⑥ Conclusiones

Ejemplos de imágenes obtenidas en cada experimento



① Introducción

② Marco teórico

③ Descripción del sistema

④ Experimentación

Aproximación 1

Aproximación 2

Experimentación en tiempo real

⑤ Resultados

⑥ Conclusiones

Experimento 1 y Experimento 2

- Las imágenes en escenario blanco se preprocesan y se entrena con ellas.

Experimento 1 y Experimento 2

- Las imágenes en escenario blanco se preprocesan y se entrena con ellas.
- Se consigue una adaptación exitosa a los datos de entrenamiento, a pesar del ruido en el pre-procesamiento de las imágenes. (En experimento 2 se cuidan más las condiciones del entorno.)

Experimento 1 y Experimento 2

- Las imágenes en escenario blanco se preprocesan y se entrena con ellas.
- Se consigue una adaptación exitosa a los datos de entrenamiento, a pesar del ruido en el pre-procesamiento de las imágenes. (En experimento 2 se cuidan más las condiciones del entorno.)
- Se observa un decaimiento en rendimiento en el conjunto de validación.

Experimento 1 y Experimento 2

- Las imágenes en escenario blanco se preprocesan y se entrena con ellas.
- Se consigue una adaptación exitosa a los datos de entrenamiento, a pesar del ruido en el pre-procesamiento de las imágenes. (En experimento 2 se cuidan más las condiciones del entorno.)
- Se observa un decaimiento en rendimiento en el conjunto de validación.
- Con las pruebas en tiempo real se identifica que al cambiar las condiciones de luz, los resultados se ven afectados, cayendo hasta un 60 % de efectividad.

Error de clasificación

Observación:

Se observó que lo que estaba clasificando el perceptrón eran las zonas donde había activaciones en lugar de los objetos que se buscaban detectar, es decir, no identificaba al objeto en sí mismo, identificaba la posición del objeto y en base a ella, determinaba la clase.

Izquierda = Auto

Derecha = Persona



Figura 9: Visualización del sesgo encontrado

① Introducción

② Marco teórico

③ Descripción del sistema

④ Experimentación

Aproximación 1

Aproximación 2

Experimentación en tiempo real

⑤ Resultados

⑥ Conclusiones

Entrenar con extracción de características

- Identificar características distintivas de los objetos a clasificar.

Entrenar con extracción de características

- Identificar características distintivas de los objetos a clasificar.
- Realizar un segundo entrenamiento con más elementos diferenciadores dentro de las imágenes de ambas clases.

Entrenar con extracción de características

- Identificar características distintivas de los objetos a clasificar.
- Realizar un segundo entrenamiento con más elementos diferenciadores dentro de las imágenes de ambas clases.
- Hacer el entrenamiento más robusto ante variaciones en el entorno.

Entrenar con extracción de características

- Identificar características distintivas de los objetos a clasificar.
- Realizar un segundo entrenamiento con más elementos diferenciadores dentro de las imágenes de ambas clases.
- Hacer el entrenamiento más robusto ante variaciones en el entorno.
- Aumentar las variaciones en ángulos, posiciones y tamaño. Así como el número de imágenes.

Entrenar con extracción de características

- Identificar características distintivas de los objetos a clasificar.
- Realizar un segundo entrenamiento con más elementos diferenciadores dentro de las imágenes de ambas clases.
- Hacer el entrenamiento más robusto ante variaciones en el entorno.
- Aumentar las variaciones en ángulos, posiciones y tamaño. Así como el número de imágenes.
- Extracción de características. Primero 18 y luego únicamente las 3 más descriptivas.(Área, longitud del área en el eje X y el eje Y).

Entrenar con extracción de características

- Identificar características distintivas de los objetos a clasificar.
- Realizar un segundo entrenamiento con más elementos diferenciadores dentro de las imágenes de ambas clases.
- Hacer el entrenamiento más robusto ante variaciones en el entorno.
- Aumentar las variaciones en ángulos, posiciones y tamaño. Así como el número de imágenes.
- Extracción de características. Primero 18 y luego únicamente las 3 más descriptivas.(Área, longitud del área en el eje X y el eje Y).
- El modelo convergió en 18,162 épocas en 1 hora 1 minuto 54 segundos. Logrando 100 % en desempeño de entrenamiento, validación y prueba.

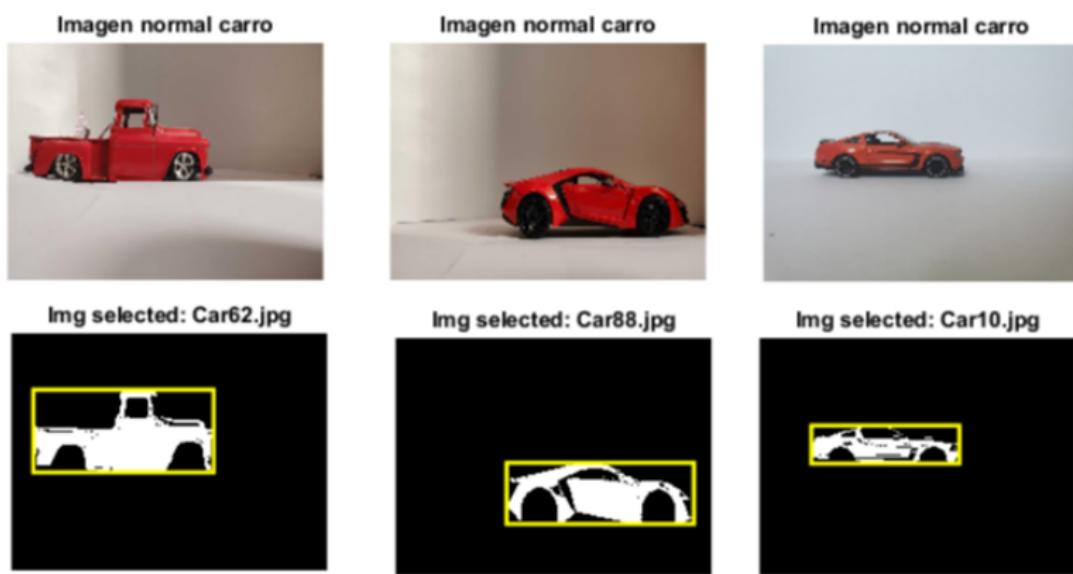


Figura 10: Ejemplos de carros encerrados en una caja a la cual se le extraen características tales como dimensiones y área.

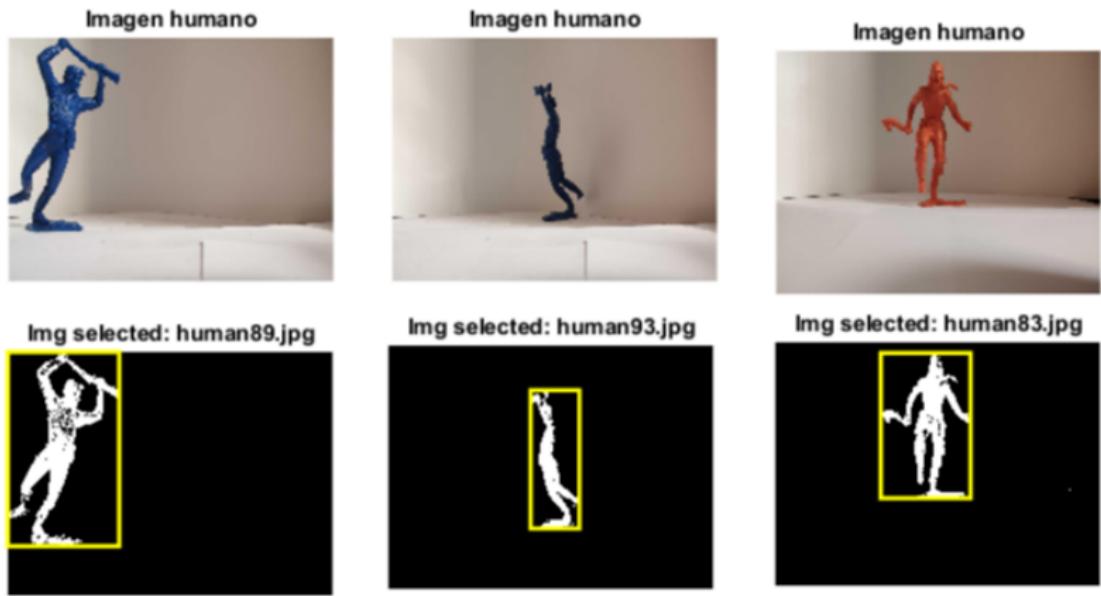


Figura 11: Ejemplos de humanos encerrados en una caja a la cual se le extraen características tales como dimensiones y área.

① Introducción

② Marco teórico

③ Descripción del sistema

④ Experimentación

Aproximación 1

Aproximación 2

Experimentación en tiempo real

⑤ Resultados

⑥ Conclusiones

Pruebas en tiempo real condiciones similares y diferentes

- Con la metodología y los pesos de la segunda aproximación, se observo 100 % de exactitud en pruebas en tiempo real con 30 imágenes por clases, realizadas de forma aleatoria.

Pruebas en tiempo real condiciones similares y diferentes

- Con la metodología y los pesos de la segunda aproximación, se observo 100 % de exactitud en pruebas en tiempo real con 30 imágenes por clases, realizadas de forma aleatoria.
- Se realizo una serie de pruebas con 15 imágenes en tiempo real manteniendo condiciones controladas pero con variabilidad alta en iluminación, cámaras de diferentes resoluciones, y posiciones de la cámara. Con ello, aun así se obtuvo 15/15 respuestas correctas.

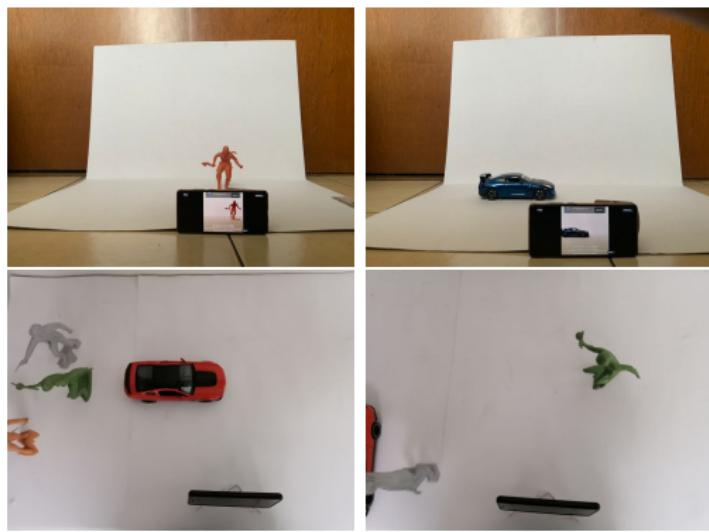


Figura 12: Representación de coches y personas en escenario controlado.

① Introducción

② Marco teórico

③ Descripción del sistema

④ Experimentación

⑤ Resultados

Métricas de evaluación

⑥ Conclusiones

① Introducción

② Marco teórico

③ Descripción del sistema

④ Experimentación

⑤ Resultados

Métricas de evaluación

⑥ Conclusiones

Métricas de evaluación: Entrenamiento

Cuadro 4: Resultados en entrenamiento.

Métricas de evaluación: Entrenamiento

	Aproximación 1		Aproximación 2
	Experimento 1	Experimento 2	
Precisión	1.000	1.000	1.000
Recuperación	1.000	1.000	1.000
F1	1.000	1.000	1.000
Exactitud	1.000	1.000	1.000
Especificidad	1.000	1.000	1.000

Métricas de evaluación: Entrenamiento

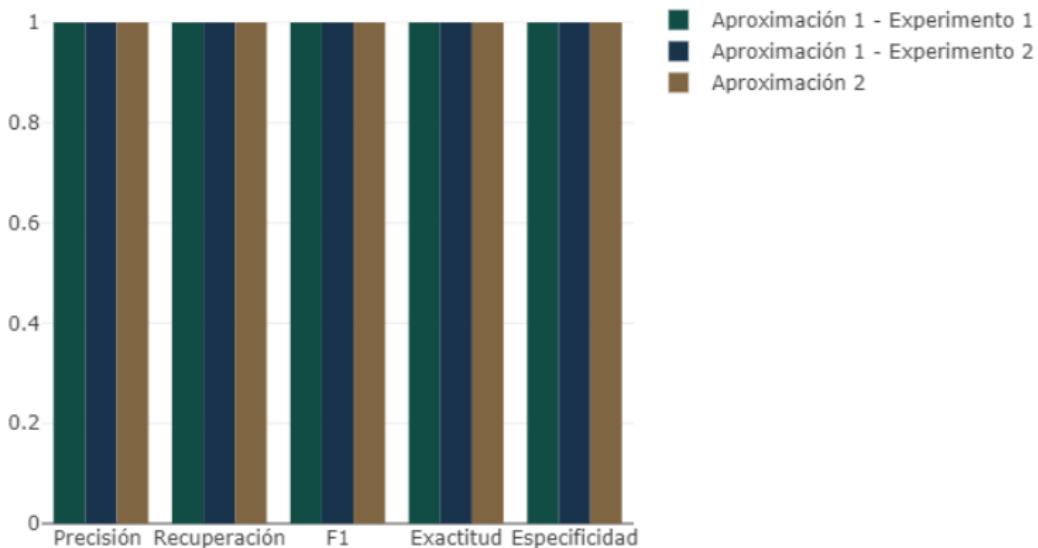


Figura 13: Resultados de desempeño en entrenamientos

Métricas de evaluación: Validación

Cuadro 5: Resultados en validación.

Métricas de evaluación: Validación			
	Aproximación 1		Aproximación 2
	Experimento 1	Experimento 2	
Precisión	0.786	0.929	1.000
Recuperación	0.917	0.867	1.000
F1	0.846	0.896	1.000
Exactitud	0.833	0.900	1.000
Especificidad	0.750	0.934	1.000

Métricas de evaluación: Validación

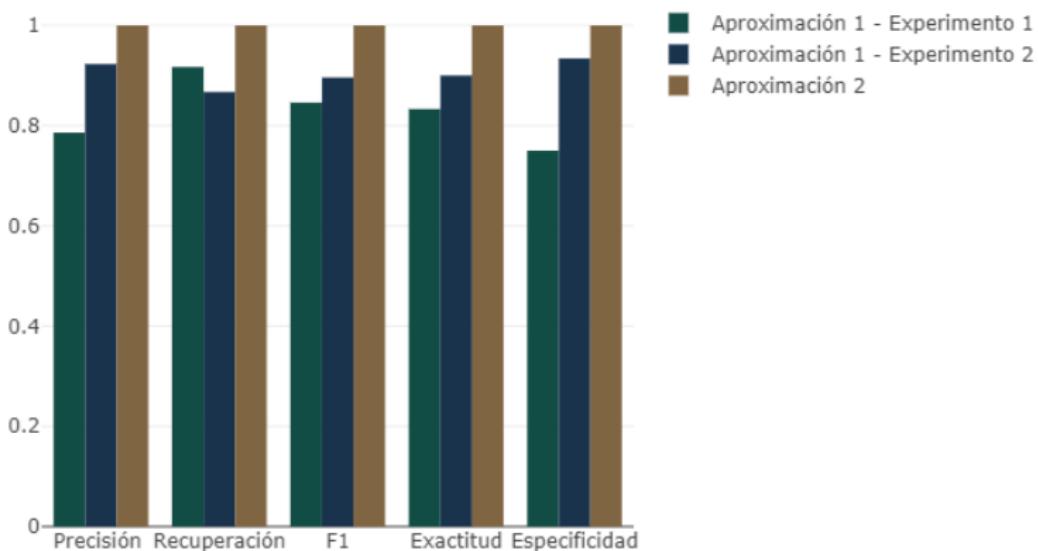


Figura 14: Resultados de desempeño en validaciones

Métricas de evaluación: Tiempo real.

Cuadro 6: Resultados en tiempo real.

Métricas de evaluación: Pruebas en Tiempo Real

	Aproximación 1		Aproximación 2
	Experimento 1	Experimento 2	
Precisión	0.583	0.636	1.000
Recuperación	0.700	0.700	1.000
F1	0.636	0.667	1.000
Exactitud	0.600	0.650	1.000
Especificidad	0.500	0.600	1.000

Métricas de evaluación: Pruebas en Tiempo Real

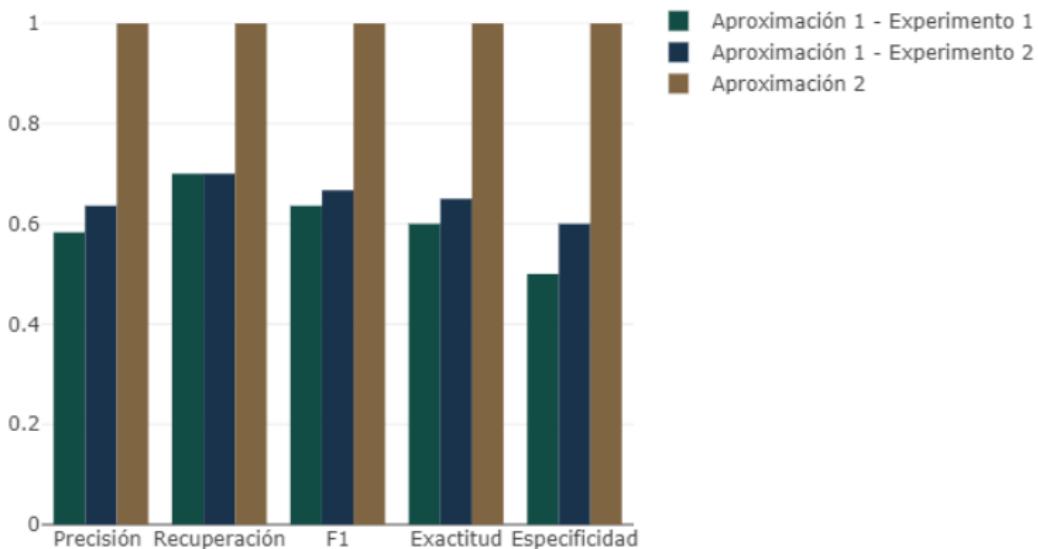


Figura 15: Resultados de desempeño en pruebas en tiempo real

Cuadro 7: Resultados en tiempo real fuera de distribución.

Métricas de evaluación: Prueba tiempo real con imágenes fuera de la distribución de entrenamiento

Aproximación 2

Precisión	1.000
Recuperación	1.000
F1	1.000
Exactitud	1.000
Especificidad	1.000
Especificidad	1.000

Prueba tiempo real con imágenes fuera de la distribución

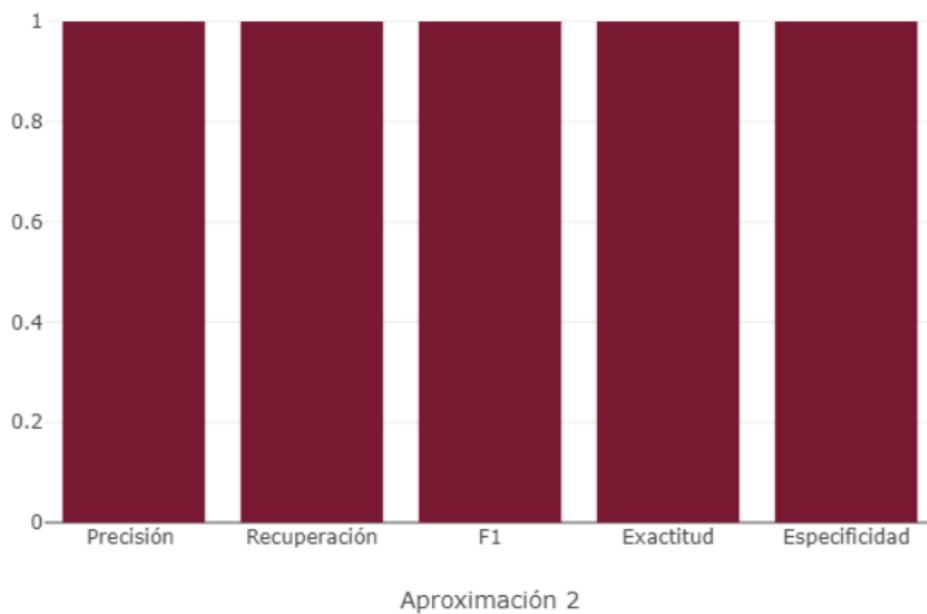


Figura 16: Resultados de desempeño en pruebas fuera de distribución

① Introducción

② Marco teórico

③ Descripción del sistema

④ Experimentación

⑤ Resultados

⑥ Conclusiones

Conclusiones

Durante el desarrollo de este proceso, al binarizar la imagen en el primer modelo experimental, el perceptrón logró identificar un patrón, sin embargo, este patrón no era el esperado. Esta situación dio como resultado una falsa interpretación de las circunstancias a evaluar por la neurona. Sin embargo, este error permitió la generación de una idea diferente para la solución del problema, en vez de considerar la imagen binaria como información suficiente, se consideró la extracción de las características de los objetos, lo que desemboco en que el perceptrón identificara correctamente una persona o un vehículo, es importante mencionar, que para la obtención de estas características no fueron necesario un cálculo o procesamiento complejo de las imágenes, ya que, según los experimentos, basta con distinguir la relación de distancia en dos dimensiones que ocupan en el espacio, así como el área de cada objeto.

- [AOB⁺18] Roni Ash, Dolev Ofri, Jonathan Brokman, Idan Friedman, and Yair Moshe.
Real-time pedestrian traffic light detection.
In 2018 IEEE International Conference on the Science of Electrical Engineering in Israel (ICSEE), pages 1–5. IEEE, 2018.
- [GECK16] Bilal Ghazal, Khaled ElKhatib, Khaled Chahine, and Mohamad Kherfan.
Smart traffic light control system.
In 2016 third international conference on electrical, electronics, computer engineering and their applications (EECEA), pages 140–145. IEEE, 2016.
- [MH14] Volodymyr Miz and Vladimir Hahanov.
Smart traffic light in terms of the cognitive road traffic management system (ctms) based on the internet of things.
In Proceedings of IEEE east-west design & test symposium (EWDTS 2014), pages 1–5. IEEE, 2014.
- [Ste90] I Stephen.
Perceptron-based learning algorithms.
IEEE Transactions on neural networks, 50(2):179, 1990.
- [TSS07] Malik Tubaishat, Yi Shang, and Hongchi Shi.
Adaptive traffic light control with wireless sensor networks.
In 2007 4th IEEE Consumer Communications and Networking Conference, pages 187–191. IEEE, 2007.
- [Wen08] W Wen.
A dynamic and automatic traffic light control expert system for solving the road congestion problem.
Expert Systems with Applications, 34(4):2370–2381, 2008.
- [WVVK04] MA Wiering, J van Veenen, Jilles Vreeken, and Arne Koopman.
Intelligent traffic light control, 2004.

Gracias por su atención

Datos de contacto:

Mario Rosas Otero

malttz15@comunidad.unam.mx

Natalia Sánchez Patiño

nataliasanchezpat@comunidad.unam.mx

Dr. David Tinoco Varela

dativa19@hotmail.com