



Universidad Autónoma de Querétaro
Facultad de Informática

Desarrollo de un algoritmo evolutivo con una codificación simbólica
para estimar el desplazamiento de objetos

Tesis

Que como parte de los requisitos
para obtener el Grado de
Maestro en Ciencias de la Computación

Presenta

I.B ANDREA ELIZABETH GÓNGORA TUN

Dirigido por:

Dra. Ana Marcela Herrera Navarro

Querétaro, Qro. a 14 de enero de 2025

La presente obra está bajo la licencia:
<https://creativecommons.org/licenses/by-nc-nd/4.0/deed.es>



CC BY-NC-ND 4.0 DEED

Atribución-NoComercial-SinDerivadas 4.0 Internacional

Usted es libre de:

Compartir — copiar y redistribuir el material en cualquier medio o formato

La licenciante no puede revocar estas libertades en tanto usted siga los términos de la licencia

Bajo los siguientes términos:



Atribución — Usted debe dar [crédito de manera adecuada](#), brindar un enlace a la licencia, e [indicar si se han realizado cambios](#). Puede hacerlo en cualquier forma razonable, pero no de forma tal que sugiera que usted o su uso tienen el apoyo de la licenciante.



NoComercial — Usted no puede hacer uso del material con [propósitos comerciales](#).



SinDerivadas — Si [remezcla, transforma o crea a partir](#) del material, no podrá distribuir el material modificado.

No hay restricciones adicionales — No puede aplicar términos legales ni [medidas tecnológicas](#) que restrinjan legalmente a otras a hacer cualquier uso permitido por la licencia.

Avisos:

No tiene que cumplir con la licencia para elementos del material en el dominio público o cuando su uso esté permitido por una [excepción o limitación](#) aplicable.

No se dan garantías. La licencia podría no darle todos los permisos que necesita para el uso que tenga previsto. Por ejemplo, otros derechos como [publicidad, privacidad, o derechos morales](#) pueden limitar la forma en que utilice el material.



Universidad Autónoma de Querétaro
Facultad de Informática
Maestría en Ciencias de la Computación

Desarrollo de un algoritmo evolutivo con una codificación simbólica para estimar
el desplazamiento de objetos

Tesis

Que como parte de los requisitos para obtener el Grado
Maestro en Ciencias de la Computación

Presenta

I.B Andrea Elizabeth Góngora Tun

Dirigido por:

Dra. Ana Marcela Herrera Navarro

Dra. Ana Marcela Herrera Navarro
Presidente

Dra. Sandra Luz Canchola Magdaleno
Secretario

Dr. Hugo Jiménez Hernández
Vocal

M. en C. Daniel Cantón Enríquez
Suplente

Dr. Jorge Domingo Mendiola Santibáñez
Suplente

Centro Universitario, Querétaro, Qro.
México
Enero 2025

DEDICATORIA

A todas las mujeres que han soñado, luchado y perseverado en el camino de la ciencia y la tecnología.

Que este trabajo sea un homenaje al esfuerzo y un recordatorio de que el conocimiento no tiene límites, ni género.

AGRADECIMIENTOS

Extiendo mi más cordial agradecimiento al Dr. Hugo Jiménez Hernández y a la Dra. Ana Marcela Herrera Navarro por su asesoría para la conclusión de este proyecto de investigación, guía académica y apoyo a lo largo de la maestría.

A los profesores de la Maestría en Ciencias de la Computación quiénes nos compartieron no sólo sus conocimientos sino también su motivación e interés en la investigación.

A mis compañeros de generación quiénes se convirtieron también en amigos.

A los compañeros de Laboratorio del CIICCTE por los gratos momentos compartidos.

A mi madre por estar siempre presente y a enseñarme el valor del esfuerzo.

Por último agradezco al Consejo Nacional de Humanidades, Ciencias y Tecnologías (CONAHCYT) por el apoyo económico brindado a través de la Beca Nacional.

RESUMEN

Una tecnología eficiente es la visión por computadora porque permite de forma no intrusiva inferir y medir variables para la toma de decisión, sin la necesidad de una instrumentación in situ sobre los escenarios. Este trabajo presenta el desarrollo de algoritmos eficientes para la medición y desplazamiento de los objetos que se monitoreen en las ciudades mediante cámaras. Este trabajo se focaliza para presentar un método eficiente en exteriores con condiciones cambiantes para el seguimiento de la dinámica de los vehículos en avenidas. Este método utiliza una propuesta basada en un algoritmo evolutivo y una codificación robusta en condiciones complicadas de cambios de luz. Las aplicaciones derivadas de esta tecnología incluyen la inferencia de la actividad en escenas, el seguimiento de vehículos, análisis de peatones y grupos entre otros, permitiendo el desarrollo de infraestructura inteligente para la medición de variables urbanas que ayuden a la toma de decisiones eficiente.

Palabras Clave: Motion estimation, evolutionary algorithm, automatic monitoring.

SUMMARY

An efficient technology is computer vision because it allows, in a non-intrusive way, to infer and measure variables for decision making, without the need for an instrumentation in situ on the scenes. This work presents the development of efficient algorithms for the measurement and placement of objects that are monitored in cities using cameras. This work focuses on presenting an efficient method outdoors under changing conditions for monitoring the dynamics of vehicles on avenues. This method uses a proposal based on an evolutionary algorithm and robust coding in complicated conditions of light changes. Applications derived from this technology include the inference of activity in scenes, vehicle tracking, pedestrian analysis, and groups among others, allowing the development of intelligent infrastructure for the measurement of urban variables that aid efficient decision making.

Keywords: Motion estimation, evolutionary algorithm, automatic monitoring.

TABLA DE CONTENIDOS

ÍNDICE DE FIGURAS.....	VII
ÍNDICE DE TABLAS.....	VIII
1 INTRODUCCIÓN.....	1
1.1 PLANTEAMIENTO DEL PROBLEMA.....	2
1.1.1 Cambios de iluminación	2
1.1.2 Desplazamientos largos	2
1.2 JUSTIFICACIÓN.....	3
1.3 OBJETIVOS	4
1.3.1 Objetivo general.....	4
1.3.2 Objetivo particulares	4
1.4 PREGUNTAS DE INVESTIGACIÓN	5
1.5 HIPÓTESIS.....	5
2 FUNDAMENTOS TEÓRICOS.....	6
2.1 REGISTRO DE IMÁGENES	6
2.1.1 Medidas de similaridad.....	6
2.1.2 Métricas de similitud	7
2.2 DETECCIÓN DE MOVIMIENTO	9
2.3 FLUJO ÓPTICO	10
2.3.1 Cambios de iluminación	12
2.3.2 Desplazamientos largos	12
2.3.3 Estimación local.....	13
2.4 ALGORITMOS GENÉTICOS	14
2.5 SISTEMAS DE CODIFICACIÓN	21

3 ANTECEDENTES	24
3.1 MÉTODOS PARA EL CÁLCULO DEL FLUJO ÓPTICO.....	25
3.1.1 Método de Horn-Schunck.....	26
3.1.2 Método de Lucas-Kanade.....	27
3.1.3 Método de Shi-Tomasi.....	28
3.1.4 Método de basado en algoritmos genéticos	29
3.2 APLICACIONES DE LOS ALGORITMOS GENÉTICOS	30
3.2.1 Internet de las cosas.....	30
3.2.2 Señalamiento inteligente del tráfico	30
4 METODOLOGÍA.....	31
4.1 ADQUISICIÓN DE SECUENCIAS DE FOTOGRAMAS	31
4.2 SELECCIÓN DEL OBJETIVO.....	33
4.3 CODIFICACIONES BINARIAS	36
4.4 BÚSQUEDA POR ALGORITMO GENÉTICO.....	40
4.5 ESTIMACIÓN DEL DESPLAZAMIENTO	40
5 RESULTADOS	42
6 CONCLUSIONES Y TRABAJO FUTURO.....	48
7 REFERENCIAS.....	49

ÍNDICE DE FIGURAS

2.1	Registro de imágenes.....	8
2.2	Cambios de iluminación en ambientes externos	13
2.3	Esquema del cruzamiento.....	21
2.4	Esquema del proceso de mutación.....	22
3.1	Ejemplo método de resolución Shi-Tomasi.....	29
4.1	Esquema de la metodología.....	32
4.2	Instrumentación.....	33
4.3	Secuencia de fotogramas.....	34
4.4	Imagen objetivo.....	35
4.5	Imagen objetivo en posición final e inicial.....	36
4.6	Imagen binarizada.....	37
4.7	Construcción de cadenas binarias.....	38
4.8	Construcción de cadenas binarias de la población.....	39
4.9	Esquema del desplazamiento.....	41
5.1	Contenido de bits que difieren de la cadena objetivo.....	43
5.2	Metodología con 30 generaciones.....	43
5.3	Resultados 10 generaciones.....	45
5.4	Resultados 50 generaciones.....	45
5.5	Resultados 100 generaciones.....	46
5.6	Resultados 500 generaciones.....	46
5.7	Resultados 1000 generaciones.....	47
5.8	Errores obtenidos.....	47

ÍNDICE DE TABLAS

5-1	Resultados de experimentos realizados	44
-----	---	----

1. INTRODUCCIÓN

Dentro del campo de visión por computadora, el análisis del movimiento y estimación, ha sido un área de investigación relevante ya que es fundamental para el avance de tecnologías como la conducción autónoma, el seguimiento de objetos, el reconocimiento de acciones, entre otras aplicaciones. El método de flujo óptico sirve para estimar el movimiento de cada píxel en una imagen generando un mapa detallado del movimiento en toda la escena. A medida que los campos de aplicación han crecido, los criterios de evaluación de estas técnicas también han aumentado, ya que se emplean en situaciones cada vez más complejas como enfrentarse a situaciones de discontinuidad del movimiento, desplazamientos largos, cambios de iluminación o costes computacionales altos y a pesar de los avances hechos abordar estos todos estos desafíos con un único método sigue siendo un problema abierto (Fortun et al., 2015).

Además existe un creciente interés por la seguridad y protección lo que esta motivando un interés sobre el análisis automatizado de video para el reconocimiento y seguimiento de objetos, además, ha aumentado el número de dispositivos utilizados para capturar imágenes, como cámaras de video vigilancia y teléfonos móviles, así obtener imágenes se ha vuelto más accesible. La mayoría de nuestros entornos, incluidas las calles, estacionamientos, escuelas y centro comerciales están monitorizadas por sistemas de video vigilancia (Kumar and Rawal, 2022).

La gestión eficaz del tráfico también tiene un campo de oportunidad en el área de visión por computadora. Debido a que el incremento de población en las ciudades trajo consigo retos en la infraestructura y movilidad, por lo que es necesario el desarrollo de nuevas tecnologías Ismagilova et al. (2019). Así el reconocimiento y seguimiento de objetos se ha vuelto un área de estudio en el campo de visión por computadora en el cual se identifican y monitorizan objetos a lo largo de secuencias de imágenes, también llamados fotogramas que posteriormente describen el comportamiento de los objetos.

1.1 Planteamiento del problema

El método de flujo óptico presenta varios retos importantes debido a que es principalmente usado para aplicaciones en el exterior como monitoreo del tráfico o conducción autónoma donde los videos captados están expuestos a cambios lumínicos debido al clima, aparición de sombras por las nubes, luz solar variable o niebla, ambientes donde principalmente la constante de la luminosidad no se cumple.

Además en el cálculo del desplazamiento, donde se emplea el método de flujo óptico, se representa como una función numérica que es aproximada mediante series de Taylor y derivadas, sin embargo, estas aproximaciones pueden generar errores en el cálculo, debido a que el método de la derivada tiende a caer en soluciones locales.

1.1.1 Cambios de iluminación

La mayoría de los métodos de flujo óptico consideran la constancia del brillo y suavidad por lo que han mostrado buenos resultados en superficies Lambertianas con iluminación constante y en objetos que se mueven en condiciones de brillo homogéneo (Shah and Xuezhi, 2021). Sin embargo, al aplicar estos métodos en entornos al exterior como en el control del tráfico y la conducción autónoma se tienen varios factores de variación relacionados con el clima, las nubes, la luz solar variable y la niebla que pueden afectar a la imagen obtenida. Se han propuesto diferentes soluciones como suponer una constancia del brillo por gradientes, sin embargo, esta suposición sólo puede manejar cambios lineales de iluminación y no funciona correctamente con grandes cambios de iluminación (Zhai et al., 2021).

1.1.2 Desplazamientos largos

Los desplazamientos a distancias grandes proporcionales al tamaño del objeto son una fuente de variación que ocurre por un objeto que se desplaza a alta

velocidad o a baja velocidad en el cambio de cuadros de vídeo consecutivos. Este problema ocurre debido a que los algoritmos de flujo óptico comúnmente asumen la constancia y suavidad del brillo dentro de una región con desplazamiento grande y como consecuencia la solución se detiene en un mínimo local incorrecto, lo cual genera errores en las iteraciones siguientes de un algoritmo (Yan and Xiang, 2016).

En las metodologías previas para estimar el desplazamiento no se considera los desplazamientos largos, ya que tienden a converger en mínimos locales, de manera que, si el desplazamiento del objeto es grande como consecuencia puede salir del rango de búsqueda. Y tampoco se toma en cuenta las variaciones de intensidad lumínica que puede haber en un video y sus imágenes secuenciales. Un cuadro de video puede presentar una intensidad lumínica mayor y el siguiente cuadro de video una menor, lo que causa un error en el resultado de la solución del desplazamiento. A pesar de las mejoras obtenidas en el algoritmo de Reynosa-Guerrero et al., 2021, una de sus limitaciones es que una región objetivo no contenga suficiente información de textura en términos de intensidades de color, es decir las diferentes intensidades lumínicas de las imágenes pueden afectar el método.

1.2 Justificación

La estimación del flujo óptico es de gran importancia para estimar el desplazamiento, y es esencial para desarrollar capacidades de comprensión de escenas como la conducción autónoma, reconocimiento de objetos, seguimiento de objetos, reconocimiento de acciones y segmentación de escenas (Zhai et al., 2021). Por esta razón es necesario un algoritmo robusto a cambios de iluminación proveniente de entornos exteriores y cálculo en los resultados que no provengan de aproximaciones numéricas de mínimos locales. Debido a que la metodología de Reynosa-Guerrero et al., 2021 muestra una mejora en la convergencia de los resultados, se propone emplear el algoritmo genético como método de búsqueda y emplear una codificación simbólica para representar las imágenes. Los esquemas de codificación pueden ser empleados para representar información según el ámbito del problema. Los

esquemas más conocidos son el binario, octal, hexadecimal, permutación, basado en valores y árbol. La codificación binaria es el esquema más simple que involucra un coste computacional pequeño de modo que cada dato codificado puede representarse como uno de dos posibles estados 1 o 0, que al concatenar varios datos permiten la construcción de cadenas binarias. En la codificación binaria, cada bit representa las características de la solución. Esto proporciona una implementación más rápida de los operadores de cruce y mutación. El flujo de bits se puede ajustar según el problema (Katoch et al., 2021). Por esta razón se propone una metodología para la estimación del desplazamiento basada en representaciones simbólicas de la imagen objetivo y referencia y el uso de algoritmos genéticos como método de búsqueda.

1.3 Objetivos

1.3.1 Objetivo general

Desarrollar un algoritmo evolutivo para estimar el desplazamiento de objetos en movimiento mediante cadenas de codificaciones simbólicas y mapas de disparidad.

1.3.2 Objetivo particulares

- Diseñar un algoritmo con un esquema de codificación simbólica en diferentes espacios de representación para las intensidades del píxel de una imagen.
- Implementar el algoritmo de representaciones simbólicas con puntos aleatorios de imágenes.
- Implementar las etapas de la metodología de algoritmos evolutivos con el algoritmo propuesto con codificaciones simbólicas para obtener puntos de desplazamiento.

1.4 Preguntas de investigación

¿El uso de un espacio métrico basado en una representación simbólica de la información de la intensidad de los píxeles permite definir criterios de similitud para la estimación del desplazamiento? ¿Los criterios de similitud basados en representaciones simbólicas pueden crear métodos invariantes a cambio lumínicos globales en el cálculo del desplazamiento?

1.5 Hipótesis

Si se usa una métrica basada en una representación simbólica resultado de la codificación de la intensidad de los píxeles, es posible construir un método de cálculo del desplazamiento tolerante a perturbaciones lumínicas globales del escenario mediante un algoritmo de búsqueda evolutivo.

2. FUNDAMENTOS TEÓRICOS

Uno de los temas de mayor interés en el campo de visión por computadora es el reconocimiento y rastreo de objetos. En la cual, se tiene la finalidad de identificar y caracterizar seguimiento de objetos a lo largo de secuencias de imágenes y con ello describir el comportamiento del objeto. Diversos sistemas de videovigilancia se encuentran en entornos como calles, avenidas, cruces, parques, centros comerciales, escuelas y hogares. Donde la detección y seguimiento de objetos es primordial en la seguridad pública, la gestión eficaz del tráfico y también se usa para identificar enfermedades en el ámbito médico (Kumar and Rawal, 2022).

2.1 Registro de imágenes

Un etapa importante en el procesamiento de imágenes es el registro de imágenes, donde se superponen dos o más imágenes de la misma escena tomada a diferentes tiempos, o en diferentes ángulos o por diferentes sensores de manera que la información final se obtiene a partir de la combinación de estas fuentes de datos (Zitová and Flusser, 2003).

Así, a partir del registro de imágenes se puede obtener información de dos imágenes consecutivas, de manera que se detecta la relación que existe entre la imagen de entrada y la de referencia. Por esta razón el registro de imágenes se puede entender como un problema de optimización (Nag, 2017). Mediante el registro de imágenes de un conjunto de imágenes consecutivas de un video se puede obtener información sobre los objetos en movimiento para su análisis.

2.1.1 Medidas de similaridad

Una medida de similaridad (en caso contrario, disimilitud) entre dos objetos o fenómenos puede cuantificar la dependencia (en caso contrario, independencia)

que existe entre ellos. En las imágenes, estos datos están representados como $X = \{x_1, \dots, x_i\}$ y $Y = \{y_1, \dots, y_i\}$, donde cada elemento de la secuencia corresponde a las intensidades de las imágenes que pueden representar un objeto o fenómeno. Una medida de similitud se considera una métrica si produce un valor más alto a medida que aumenta la dependencia entre los valores de la secuencia (Goshtasby, 2012).

2.1.2 Métricas de similitud

Una métrica de similitud se considera una métrica si produce un valor más alto a medida que los valores correspondientes en X y Y se vuelven menos correspondientes. Una métrica corresponde a una función de distancia que existe entre dos patrones y tiene cuatro propiedades siguientes (Duda et al., 2000):

1. No-negatividad: $D(X, Y) \geq 0$
2. Reflexividad: $D(X, Y) = 0$ si y sólo si $X = Y$
3. Simetría: $D(X, Y) = D(Y, X)$
4. Desigualdad de triángulos: $D(X, Z) + D(Z, Y) \geq D(X, Y)$

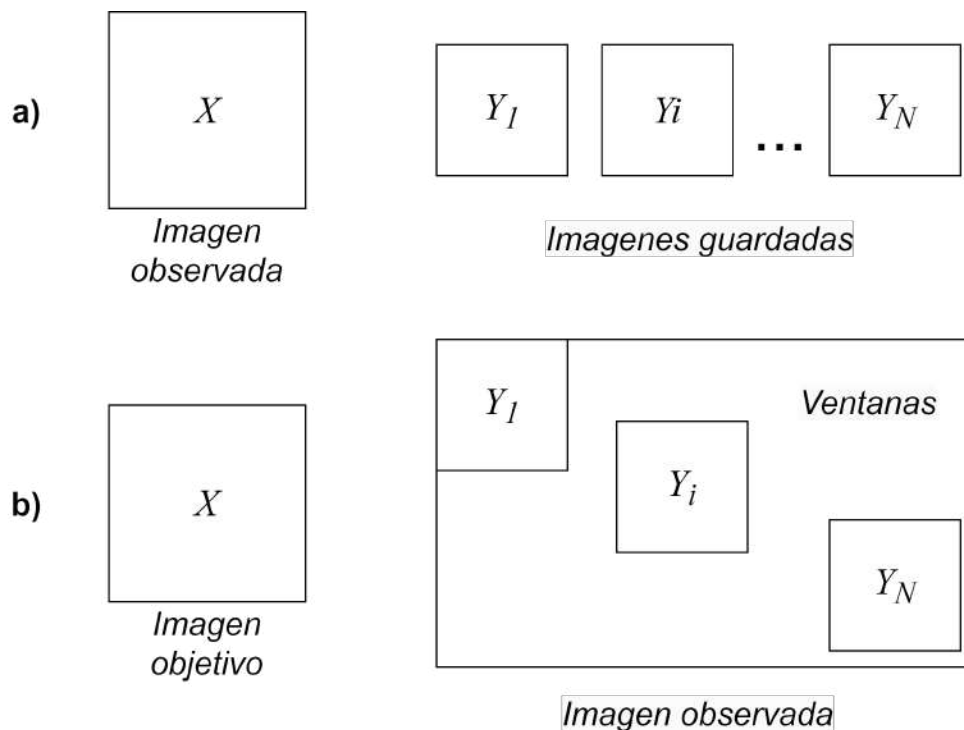
Así una métrica de similaridad D , satisface los anterior para todas las cadenas de secuencias X y Y .

Se han desarrollado diferentes medidas de similitud que indica el grado de semejanza o disimilitud si se cuantifican las diferencias, cada una con ventajas y desventajas particulares. Algunas métricas emplean directamente los valores de intensidad originales, mientras que otras ajustan o normalizan estas intensidades antes de analizarlas, mientras que otras utilizan los rangos de intensidades y en última instancia otras métricas emplean las probabilidades conjuntas de las intensidades correspondientes entre imágenes.

Las medidas de similitud/disimilitud pueden tener dos contextos de aplicación. En el primer contexto, se tiene una imagen y un conjunto de imágenes previamente almacenadas, donde se tiene como objetivo identificar la imagen almacenada que mejor corresponde con la observada, por ejemplo si se comparan imágenes de una base de datos y la observada corresponde a la imagen de una cámara. El segundo contexto de aplicación se centra en encontrar un objeto de interés en una imagen observada en la que el modelo del objeto se da en forma de plantilla y la imagen observada puede provenir de una cámara donde el objetivo es determinar la posición en la imagen donde la plantilla encaja mejor, indicando así la posición del objeto (Goshtasby, 2012). En la Figura 2.1 se ilustran ambos contextos de aplicación.

Figura 2.1

Registro de imágenes.



Nota: a) Contexto de aplicación cuyo objetivo es identificar la mejor correspondencia de un conjunto de imágenes almacenadas. b) Contexto de aplicación de búsqueda de un objetivo dentro de un conjunto de ventanas de una imagen observada. Adaptada de Goshtasby (2012).

2.2 Detección de movimiento

Una cámara puede obtener una secuencia de imágenes consecutivas, conocidas como fotogramas, las cuales conforman un video (Kulchandani and Dangarwala, 2015). Así, durante el análisis de la actividad de un video, se siguen tres etapas fundamentales:

1. Detección del objeto en movimiento
2. Seguimiento del objeto dentro una serie determinada de fotogramas
3. Analizar el objeto en movimiento para determinar su comportamiento

Se han propuesto diferentes técnicas para la detección de movimiento, las cuales pueden ser clasificadas como: sustracción de fondos, diferencia de fondos, diferencias temporales y el método de flujo óptico. Se debe tomar en cuenta que existen diferentes retos que dificultan la detección como los cambios de iluminación, el efecto de una sombra, escenas dinámicas con fondo variable, entre otros (Manchanda and Sharma, 2016).

El método de sustracción de fondo se basa en obtener un modelo de referencia de fondo que es comparado con un fotograma de video para determinar una posible variación, de manera que el movimiento se detecta por la diferencia entre un fotograma actual y el modelo de referencia de fondo. El método de diferencia de fondo identifica al objeto en movimiento calculando la diferencia de una imagen actual menos una imagen anterior; para entornos dinámicos, el resultado de la detección de objetos en movimiento resulta no preciso ya que es difícil obtener un contorno completo de ese objeto (Balaji and Karthikeyan, 2017).

El enfoque de diferencias temporales se basa en la diferencia de píxeles de los fotogramas consecuentes. Una de las desventajas del enfoque es que no proporciona buenos resultados cuando el objeto a detectar se desplaza rápido o lento debido a que hay una diferencia muy pequeña entre los cuadros de video

y en consecuencia se pierde parte de la información del video. Mientras que en el enfoque de diferencia de fondos consiste en la diferencia acertada de píxeles en los cuadros. No obstante, no provee buenos resultados cuando el objeto a ser detectado se mueve muy rápido o muy lento ya que habría una pequeña diferencia en los cuadros y el método no pudiera detectar un cambio significativo (Jusman et al., 2020)

Los métodos de seguimiento de objetos proporcionan una descripción del movimiento, pero carecen de la precisión y la disposición espacial de los estimadores de flujo óptico. Por esta razón, el método de flujo óptico es ampliamente usado para la estimación del vector de movimiento (Pérez-Rúa et al., 2016).

2.3 Flujo óptico

Cuando las escenas contienen objetos en movimiento existen variaciones en la intensidad entre los pares de imágenes consecutivas lo que permite obtener información del movimiento en la escena y de este modo se considera que las regiones con intensidad constante se encuentran sin movimiento. Esta variación es estudiada como flujo óptico, donde se aplica un operador a todos los píxeles de la imagen para generar un campo vectorial de movimiento de toda la imagen (Davies, 2018). El método de flujo óptico, se refiere a los desplazamientos que existen en los patrones de intensidad en el plano de imagen, en la práctica estos vectores de desplazamiento representan el desplazamiento real de los objetos. No obstante, este método supone la constancia del brillo, incluyendo los objetos que no están en movimiento. Los cambios de intensidad no son necesariamente causados por los desplazamientos de los objetos, si no que también se pueden deber a otros fenómenos como los cambios de iluminación, efectos de reflexión o modificaciones de las propiedades de los objetos que afectan su emisión de luz o reflectancia (Fortun et al., 2015).

La estimación se realiza, dada una secuencia de imágenes denotada por $I : \Omega \times T \rightarrow R$, donde $\Omega \subset R^2$ es el dominio de la imagen y T , es el intervalo de tiempo de la secuencia. Para la estimación del desplazamiento, los métodos se basan en la suposición de la relación ente el campo de movimiento buscado $w : \Omega \subset R^2$, en el tiempo t de la imagen $I(:, t)$. La suposición de que la intensidad de los píxeles se mantiene constante durante el desplazamiento es ampliamente utilizada y es representada mediante la ecuación 2.1.

$$\frac{dI}{dt}(x(t), t) = 0 \quad (2.1)$$

La aproximación discreta de la ecuación 2.1 por cada píxel dado un tiempo determinado se muestra en la ecuación 2.2. Sin embargo, para resolver esta ecuación como un modelo de optimización se emplea la versión extendida con derivadas parciales, resultando en la ecuación 2.3

$$I(x + w(x), t + 1) - I(x, t) = 0 \quad (2.2)$$

$$\frac{\partial I}{\partial x_1}(x)u(x) + \frac{\partial I}{\partial x_2}(x)v(x) + \frac{\partial I}{\partial t}(x) = 0 \quad (2.3)$$

No obstante, el método enfrenta algunos desafíos, uno de ellos radica en que no todos los movimientos provocan variaciones en los niveles de iluminación en una imagen donde no todas las variaciones de intensidad en los píxeles son causadas únicamente por el movimiento de objetos. Este problema puede ser causado por cambios de brillo o también por observaciones de objetos transparentes o no lambertianos. Otras causas en los desafíos se deben al ruido integrado en el sensor de la cámara, sobre todo en malas condiciones de iluminación. Otro reto proviene del llamado problema de apertura. Este problema ocurre cuando se observa un ob-

jeto cuya textura no varía y en caso de que el objeto sin textura se mueva dentro de la imagen, se necesitará información adicional para recuperar el movimiento (Kajo et al., 2015). En la sección 2.3.1 se detallan los problemas causados por los cambios de iluminación, en la 2.3.2 los desplazamientos largos y en la 2.3.3 se explicarán las desventajas que tiene el uso de métodos numéricos.

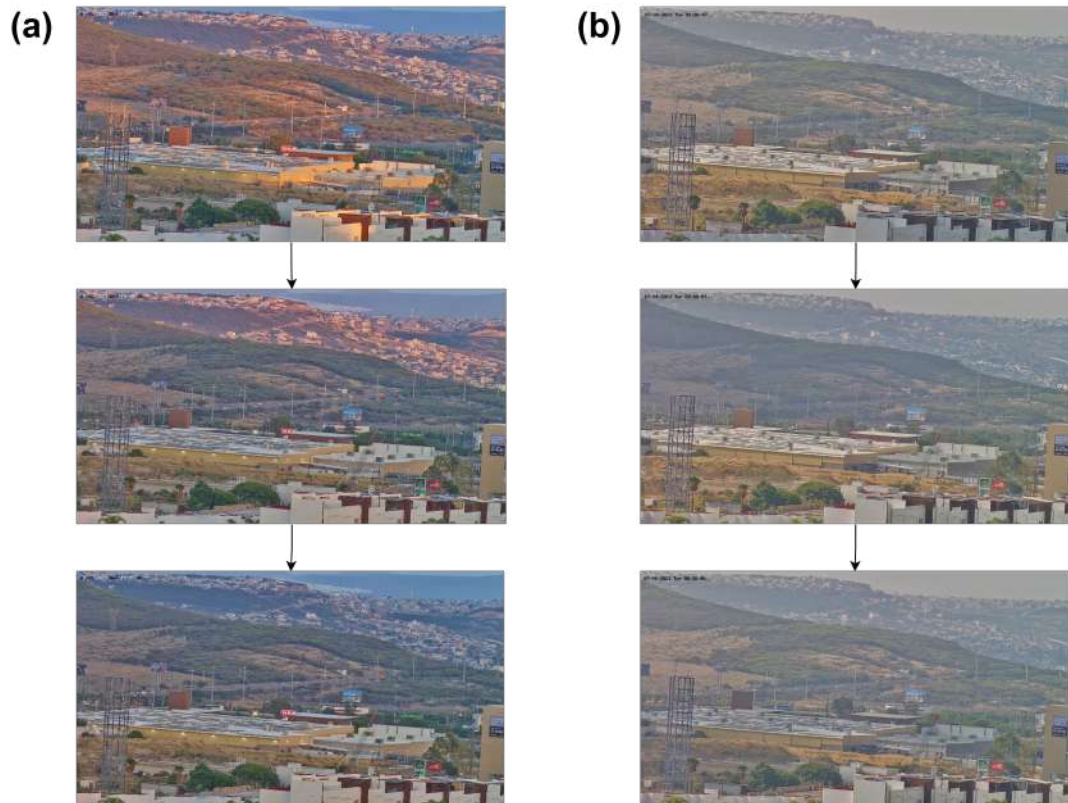
2.3.1 Cambios de iluminación

Uno de las problemáticas del flujo óptico es su sensibilidad a las variaciones lumínicas de una imagen. La mayoría de los métodos de flujo óptico asumen que la intensidad de los píxeles es constante entre fotogramas consecutivos. Por lo que variaciones en la iluminación, como sombras causadas por el movimiento, reflejos, cambios en la luz solar o la luz artificial, afecta significativamente el cálculo del desplazamiento. Ya que estas alteraciones conllevan a estimaciones incorrectas del desplazamiento, debido a que el método de flujo óptico interpreta erróneamente los cambios de brillo como desplazamientos.

2.3.2 Desplazamientos largos

Los desplazamientos largos de un objeto en la imagen también generan problemáticas en el cálculo del desplazamiento y es causante de que varios algoritmos de flujo óptico fallen. Los desplazamientos largos ocurren como resultado de un objeto que se mueve a alta velocidad o a una baja velocidad en el cambio de cuadros de video consecutivos. Los algoritmos para la estimación del desplazamiento no abordan desplazamientos largos ya que los resultando tienden a queda atrapados en mínimos locales incorrectos. Para abordar esta problemática se han propuesto diversas soluciones que se basan en una mayor cantidad de iteraciones lo que ha provocado resultados más erróneos. Para trabajar con desplazamientos largos se han propuesto esquemas *coarse-to-fine*, dentro de los trabajos realizados, el más destacado es el hecho por T.Brox (Shah and Xuezhi, 2021), en el cual se emplean diferentes descriptores para los objetos, sin embargo incrementa el costo

Figura 2.2
Cambios de iluminación en ambientes externos



Nota: Cambios de iluminación causado por a) Cambio diurno-nocturno y b) Condiciones climáticas (lluvia).

computacional.

2.3.3 Estimación local

Los métodos para la estimación del flujo óptico pueden usar técnicas globales y locales. Las primeras se basan en las consideraciones de la constancia del brillo y la suavidad de una imagen, donde se construye una función de energía cuyo esquema de minimización produce el campo de flujo óptico. Quienes iniciaron este enfoque fueron Horn y Schunck, y posteriormente varios investigadores siguieron su trabajo, sin embargo el algoritmo proporciona dificultades en situaciones prácticas donde el método es expuesto a entornos externos. Por otra parte, los enfoques locales parten del un flujo óptico constante dentro de un pequeño vecindario de la imagen que constan de $n \times n$ píxeles, donde la ecuación del flujo se evalúa en to-

dos los píxeles dentro de la ventana del vecindario y las ecuaciones resultantes se resuelven utilizando el método de mínimos cuadrados (Shah and Xuezh, 2021).

La derivación del flujo óptico se basa en una expansión de la serie de Taylor de primer orden, sin embargo, esto es válido si el movimiento entre dos sistemas es pequeño. Del mismo modo, las ecuaciones están escritas para variables cuyo tiempo es de espacio continuo y se tiene que estimar el gradiente de la imagen, por lo que será necesario discretizar estas variables, este tipo de muestreo introduce errores en la solución (Pesquet-Popescu et al., 2013). Las desventajas de los métodos numéricos incluyen la posibilidad de errores de aproximación debido a la necesidad de realizar aproximaciones y simplificaciones, lo que puede resultar en soluciones inexactas. Existe una representación finita de los números en la computadora lo que puede causar problemas de precisión. Además ciertos métodos necesitan varias iteraciones para converger, especialmente en problemas mal condicionados o funciones complicadas. La sensibilidad a las condiciones iniciales, donde pequeños cambios pueden conducir a resultados significativamente diferentes también son unas de sus problemáticas. Así como las limitaciones en problemas no lineales o mal condicionados, que pueden reducir la eficiencia y la precisión de los métodos numéricos.

2.4 Algoritmos genéticos

La computación evolutiva incluye todas las metodologías de resolución de problemas inspirados en los procesos evolutivos genéticos naturales. Dentro de sus características lo hacen robusto se encuentra que se pueden implementar de manera paralela lo que permite explorar el uso de otras arquitecturas *multi-core* y *many-core*. Del mismo modo, sus características relacionadas con las propiedades de convergencia, optimización de parámetros y su espacio de búsqueda de soluciones (provenientes de la función de aptitud), entre otras (Cicirello, 2024).

A este tipo de procedimientos se les conoce como algoritmos metaheurísti-

cos, y existen dos tipos, los basados en soluciones únicas y los basados en poblaciones. Los algoritmos basados en soluciones únicas emplean una solución candidata y mejoran la solución mediante una búsqueda local. Sin embargo, las soluciones obtenidas pueden estancarse en óptimos locales. Los segundos basados en poblaciones, emplean múltiples soluciones candidatas durante el proceso de búsqueda, estas metaheurísticas mantienen la diversidad en la población y evitan que las soluciones se detengan en mínimos locales (Katoch et al., 2021).

Dentro de estos algoritmos se encuentra el algoritmo genético (AG), el cual está inspirado en el proceso biológico de evolución, y emula la teoría darwiniana de la supervivencia del más apto en la naturaleza (Katoch et al., 2021).

Este algoritmo fue propuesto por Holland (1992) en donde se utiliza el concepto de supervivencia del más apto. Las nuevas poblaciones se producen mediante el uso iterativo de operadores genéticos sobre los individuos presentes en la población. La representación de los cromosomas, la selección, el cruce, la mutación y el cálculo de la función de aptitud son los elementos clave de los AG. Las operaciones de selección, cruce y mutación se repetirán en la población actual hasta completar la nueva población.

Los AG cambian dinámicamente el proceso de búsqueda mediante las probabilidades de cruce y mutación llegando a la solución óptima. Los AG pueden modificar los genes codificados y pueden evaluar múltiples individuos y producir múltiples soluciones óptimas. Por lo tanto, lo que hace que los AG tengan mejor capacidad de búsqueda global (Katoch et al., 2021).

Estos algoritmos parten de una población de soluciones aleatorias, donde en cada solución se evalúa mediante una función que indica qué tan adecuada es una solución. Después de varias iteraciones, se pueden elegir las mejores soluciones. Las mejores soluciones se combinan para producir el siguiente conjunto de soluciones. De manera que los algoritmos evolutivos tienen varios componentes que seleccionan y combinan soluciones en cada población. A diferencia de los algoritmos deterministas (como la búsqueda por fuerza bruta) encuentran la misma

solución en cada ejecución, sin embargo las soluciones se obtienen de soluciones locales (Mirjalili et al., 2020). Cada solución de un problema de optimización esta representado como cromosomas y cada gen corresponde a una variable del problema. Por ejemplo si un problema tiene 10 variables, el AG empleará cromosomas con 10 genes. De este modo, en los algoritmo genéticos, el primer paso es formular el problema y definir sus variables como un vector. En referencia a los cromosomas usados puede haber dos tipos de Algoritmos genéticos: binarios y continuos. En la variante binaria cada gen puede tener uno de dos variables posibles como 0's y 1's. En el caso continuo, los genes pueden tomar cualquier valor definido por límites superior e inferior . Una variante especial de la binaria es donde existen do o más variables, en este caso, será necesario el uso de más bits para guardar las variables del problema, de este modo, el número de genes (bits) necesarios (Mirjalili et al., 2020) es calculado mediante la ecuación 2.4.

$$nbits = \log_2 n \quad (2.4)$$

Los algoritmos genéticos tienen tres principales operadores, los cuales a saber son: selección, combinación y mutación. Antes de pasar a los pasos de los operadores genéticos, el proceso de optimización empieza generando un conjunto aleatorio de cromosomas. En el caso de los algoritmos genéticos binarios, se sigue la ecuación 2.5, donde X_i es el i-ésimo gen y r_i es un número aleatorio entre 0 y 1, generados con cada gen.

$$X_i = \begin{cases} 1 & \text{si } r_i < 0.5 \\ 0 & \text{de otro modo} \end{cases} \quad (2.5)$$

Para el caso continuo, se emplea la ecuación 2.6, donde ub_i es el límite superior del i-ésimo gen y lb_i es el límite inferior del gen o variable (Mirjalili et al., 2020).

$$X_i = (ub_i - lb_i) \cdot r_i + lb_i \quad (2.6)$$

Estas variables aleatorias distribuidas uniformemente, se emplean en los siguientes pasos de los algoritmos genéticos, los cuales son llamados operadores genéticos y se describen en la siguiente sección.

Inicialización

La población inicial esta conformada por un conjunto de soluciones candidatas, donde el AG utiliza la población para evaluar las diversas áreas del espacio de búsqueda que facilitan la exploración global, por lo tanto la calidad de la población inicial puede influir en el desempeño del AG. Principalmente dos características controlan la calidad de la población, la diversidad de la población y el tamaño de la población. La diversidad de la población ayuda al AG a evaluar diferentes regiones del espacio de búsqueda, reduciendo así la probabilidad de quedarse en el óptimo local de un valor incorrecto de aptitud. Por el contrario una población con una baja tasa de diversidad conduce a un AG como un algoritmo de búsqueda local con una sobre carga de iteraciones produciendo una convergencia prematura, es decir una población contiene varios individuos similares, sin antes haber explorado el espacio completo de soluciones. Cuando se tiene un tamaño de población fijo, se puede afectar negativamente el rendimiento del algoritmo genético, ya que la probabilidad de cubrir espacios favorables del espacio de búsqueda disminuye a medida que aumenta la dimensionalidad del espacio de búsqueda. Por ello, la selección de un tamaño de población pequeño reduce la diversidad de la población, no obstante un tamaño de población grande puede desperdiciar recursos computacionales (Alhijawi and Awajan, 2023).

Función de aptitud

La función de aptitud en un algoritmo genético es utilizado para medir el valor de aptitud de los individuos. La función objetivo, es la función que mide el desempeño de los individuos en relación a un conjunto de parámetros y puede ser única

o de una función multiobjetivo. Mientras que la función de aptitud mide si un individuo será reproducido de acuerdo a lo obtenido en la función objetivo. Dependiendo de lo que se quiera resolver, se puede utilizar la función objetivo para minimizar o maximizar una única función objetivo (Alhijawi and Awajan, 2023).

Operadores Genéticos

I. Selección

Este operador tiene como objetivo seleccionar las soluciones adecuadas de una población, es decir soluciones que cumplen con un rango aceptable de la función objetivo y eliminar las que no cumplan la función objetivo. Y también se crean nuevas soluciones a partir de las mejores soluciones. Por esta razón en este operador, se ejecuta tanto una reproducción como la selección. Para saber si una solución es adecuada, en este proceso, se toma en cuenta el resultado de la función de aptitud. En donde la solución que tenga una mejor aptitud debe ser seleccionada y reproducida con otras de adecuada aptitud.

Existen diferentes métodos de selección recopilados de (Santhi and Vinodhini, 2021), listados en el siguiente párrafo :

- Método de selección aleatoria: En este tipo de selección, se eligen aleatoriamente los progenitores para llevar a cabo la operación de cruce, utilizando un generador de números aleatorios uniformes.
- Selección de ruleta: En esta estrategia, los padres se seleccionan de la población dada, basándose en la probabilidad de su valor de aptitud de cada elemento. La ventaja del método es que el cromosoma que con un valor mayor de aptitud tiene muchas probabilidades de ser el progenitor de la solución.
- Muestreo Universal Estocástico: Este método es similar a la selección de la rueda de la ruleta, en la que sólo tiene un punto fijo, mientras que en este método tiene más de uno (múltiples) puntos fijos. Este método se considera justo porque da una oportunidad de selección a los miembros más débiles de

la población.

- Selección por rango: en esta técnica, cada cromosoma recibe su rango clasificándolo en función de su valor de aptitud. El cromosoma con el valor de aptitud más alto se clasifica como N (siendo N el número de cromosomas de la población) y el siguiente valor más alto se clasifica como $N - 1$, etc. El cromosoma con el valor de aptitud más bajo se clasificará como 1. Este sistema de clasificación lleva a que cada cromosoma reciba su rango. El cromosoma con el menor valor de fitness se clasificará como 1. Este sistema de clasificación hace que cada individuo tenga una parte casi igual en el gráfico circular, como se muestra en la figura. Así, cada cromosoma tiene aproximadamente la misma probabilidad de ser seleccionado como progenitor.
- Selección por torneos: Esta estrategia de selección se utiliza para seleccionar el cromosoma más apto de la población actual. Los candidatos seleccionados pasan a la siguiente generación. La selección por torneo inicia eligiendo los k individuos de la generación y realizar un torneo entre ellos y de los k individuos se selecciona el mejor. El ganador del torneo es el cromosoma con mayor aptitud. Finalmente se repiten los pasos anteriores hasta alcanzar la cantidad de población deseada.
- Selección elitista: En esta estrategia de selección sólo se considera un número limitado de individuos con los mejores valores de fitness y pasan a la siguiente generación, evitando los operadores de cruce o mutación de individuos con adecuada genética.
- Selección por truncamiento: En este método, los cromosomas se clasifican en función de su valor de fitness. Sólo se seleccionan los mejores individuos para la reproducción de la nueva población. Los padres se seleccionan en función del valor umbral. El valor umbral tiene como finalidad seleccionar los progenitores en la proporción especificada 50% - 10% de su población. Los cromosomas que tienen valores inferiores al valor umbral no producen nueva población y se descartan.

II. Cruzamiento

El cruce permite la reproducción en los algoritmos evolutivos, asemejando lo que ocurre en la naturaleza, cuando un macho y una hembra se aparean y producen descendencia aunque algunos organismos hermafroditas, como los caracoles que pueden desempeñar el papel de madre o padre difiere significativamente en la de los algoritmos. En los algoritmos evolutivos, se permite que cualquier individuo se cruce con cualquier otro individuo, aunque en otras publicaciones se ha empleado un grupo para realizar el cruce de las cadenas (Heaton, 2014).

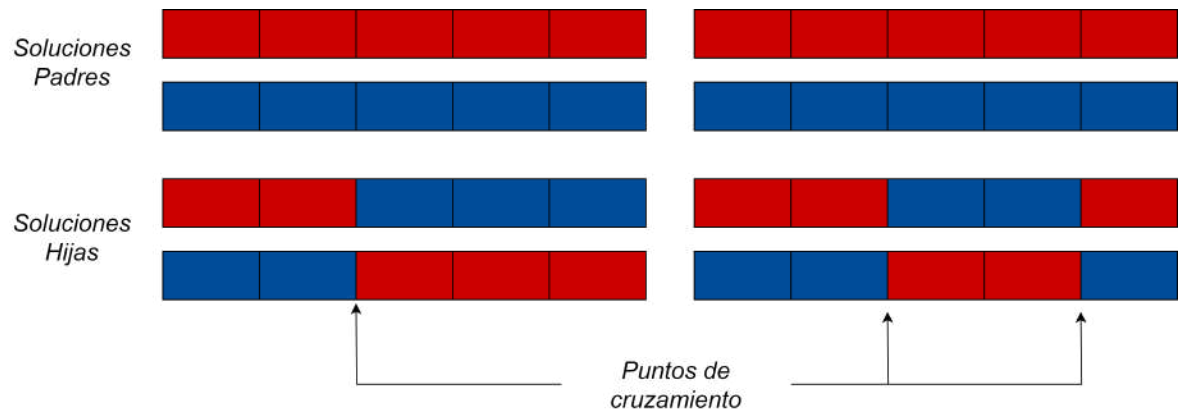
La principal razón del uso del cruzamiento es para generar nuevas soluciones a partir de soluciones previas con un alto valor de aptitud, por lo que idealmente lo que nos gustaría hacer es crear un nuevo individuo que combine las mejores características de cada uno, de este modo, se recombinan estas características al azar. La población tiene que cumplir con varios criterios listados por Alhijawi and Awajan (2023), los cuales a saber son:

- Aleatoriedad: El proceso de inicialización tiene que producir individuos impredecibles y irregulares, por lo que la inicialización sigue técnicas basadas en aleatoriedad, las cuales son, técnicas estocásticas que generan diferentes poblaciones en función a una semilla inicial. Por el contrario, la técnica determinista genera la misma población independientemente de la semilla inicial.
- Criterios de composicionalidad: Los enfoques basados en composicionalidad se pueden clasificar en dos tipos según la cantidad de procedimientos independientes involucrados en el proceso de inicialización. El primer tipo, la técnica no composicional es la técnica de inicialización más simple que produce poblaciones en un único paso. Y la segunda, la técnica composicional que utiliza más de un paso para producir la población inicial.
- Generalidad: Las técnicas de inicialización pueden dividirse en dos tipos según su generalidad: genérica y específica. En la genérica, la inicialización no depende del dominio mientras que en la específica se inicializa la población de un

problema específico.

Figura 2.3

Esquema del cruzamiento.



III. Mutación

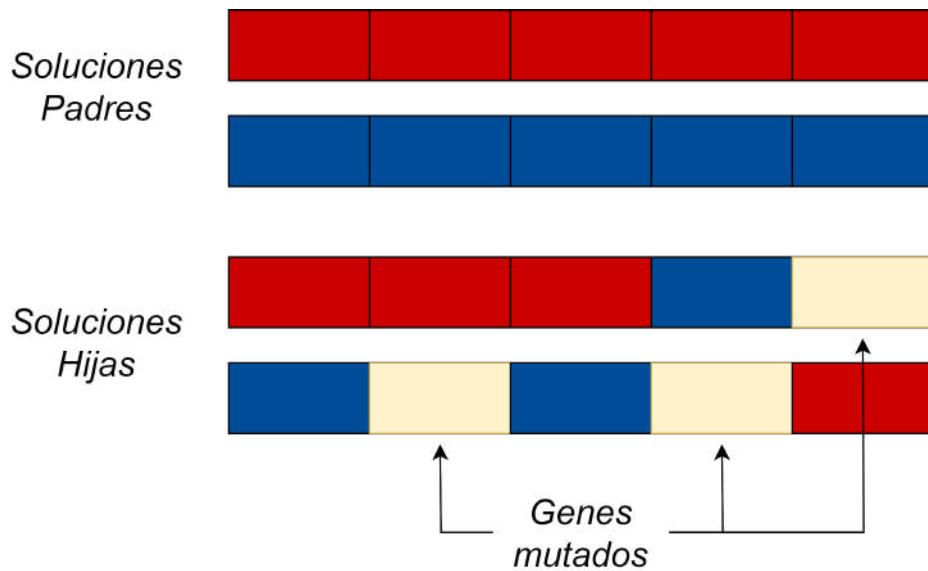
Durante el cruzamiento, se cambian genes entre cromosomas, sin embargo, esto tiene un problema debido a la carencia de nuevos genes, es decir, nuevas soluciones. Debido a que si todas las soluciones son similares, se quedan atrapadas en una solución localmente óptima. Para abordar este problema se tiene el operador de mutación (Mirjalili et al., 2020). Por esto, todos los algoritmos evolutivos funcionan combinando la selección con un mecanismo para producir variaciones. El mecanismo más conocido para producir variaciones es la mutación, en la que un alelo de un gen se sustituye aleatoriamente por otro. En otras palabras, se crean nuevas soluciones de prueba realizando pequeños cambios aleatorios en la representación de las soluciones de prueba anteriores. Si se utiliza una representación binaria, la mutación se consigue cambiando bits al azar. Un índice de mutación comúnmente utilizado es uno sobre la longitud de la cadena (Baeck et al., 2000).

2.5 Sistemas de codificación

Las imágenes pueden ser codificadas mediante cadenas binarias, de manera que, una cadena de bits puede representar en conjunto las características de una imagen. (Carlson, 1991) Los datos se pueden comprimir para representar la

Figura 2.4

Esquema del proceso de mutación.



misma información en menor procesamiento. Estas cadenas pueden ser comparadas para obtener una regla de comparación dentro de los algoritmos genéticos. (Choi et al., 2009).

Enfoque simbólico

La representación de la información puede clasificarse en varias categorías. Estrictamente local, distribuido, local, por micro-características y *coarse coding* (codificación burda). A diferencia de las clasificaciones anteriores, el uso de representaciones distribuidas, puede eliminar inconvenientes como la insuficiencia de representación para grandes conjuntos de datos, proliferación en redes que representan una estructura compleja, el uso ineficiente y redundante de conexiones, y la incertidumbre sobre su capacidad de aprender representaciones elaboradas de estructuras complejas (Mittal and Nag, 2019).

Las representaciones distribuidas, codifican un concepto específico mediante un patrón único de actividad dentro de un grupo de unidades. Y se definen en el contexto de un sistema conexionista como una representación en la que cada entidad se representa mediante un patrón de actividad distribuido en muchos elementos y cada elemento participa en la representación de muchas entidades di-

ferentes”(Carlson, 1991).

El enfoque simbólico representa la información mediante símbolos y sus relaciones; también se infieren nuevos conocimientos mediante el procesamiento de éstos símbolos. El término acuñado para estos enfoques son las Arquitecturas Simbólicas Vectoriales. Una clase de específica de las Arquitecturas Simbólicas Vectoriales son los Códigos binarios dispersos. Los cuales representan un concepto ya sea un atributo, una característica o un elemento de un conjunto, como vectores aleatorios binarios (Haq and Caballero, 2021).

Estos vectores dispersos de componentes de bajo nivel se utilizan para crear conceptos de un nivel superior al sumarse formando la superposición. Este vector denso se crea mediante la operación lógica OR en donde sus componentes corresponden a los elementos de bajo nivel. Los códigos dispersos representan partes de grandes patrones de bits. No existe una manera sencilla de extraer los componentes de un vector compuesto; sin embargo, dados dos vectores se puede medir su distancia y ver si una de ellas es componente de la otra (Kleyko, 2016).

Una ventaja de usar un esquema de codificación binario, es que los cromosomas de los algoritmos genéticos se componen de 0's y 1's , y pueden resolver directamente optimizaciones binarias (Pan et al., 2022).

3. ANTECEDENTES

El flujo óptico calcula el campo de movimiento bidimensional de la imagen, por lo que ha sido ampliamente empleado como método para la estimación del desplazamiento. Su importancia recae en el hecho de que es esencial para desarrollar capacidades de comprensión de escenas como la conducción autónoma, reconocimiento de objetos, seguimiento de objetos, reconocimiento de acciones y segmentación de escenas (Zhai et al., 2021).

Principalmente el método de global variacional de Horn-Schunk y el método de mínimos cuadrados de Lucas-Kanade han sido aplicados para el cálculo del flujo óptico. Sin embargo también se han desarrollado otras metodologías para el cálculo del flujo óptico. En el trabajo de Pan et al. (2020) se emplearon cámaras de eventos para estimar el flujo óptico incluso con imágenes borrosas, esto con el objetivo de manejar casos altamente dinámicos. En otra implementación realizada por Hu et al. (2021) se emplea una cámara de picos (*spiking camera*) donde se propone un método para estimar el flujo óptico en escenas de alta velocidad y usando una arquitectura de red neuronal modificada y emplear ventanas adaptativas del flujo. También se ha explorado el uso de redes neuronales para el cálculo. En el trabajo propuesto por Artizzu et al. (2021), se presenta la estimación del desplazamiento por medio del flujo óptico en imágenes omnidireccionales, construida sobre una red neuronal convolucional y se adapta a distorsiones equirectangulares mediante la convolución consciente de la distorsión. Otros enfoques para el cálculo del flujo óptico que se basan en la regularización del dominio de la frecuencia para realizar la estimación, fueron realizados por Chen et al. (2021). En este trabajo se hace la transformada wavelet para el campo de flujo óptico, donde se eliminan los valores atípicos.

Dentro de las aplicaciones del método de flujo óptico se encuentra su uso en navegación de robots aéreos no tripulados, esta aplicación tiene el reto de sus elevados requisitos computacionales para el procesamiento de imágenes en tiempo

real y recursos de masa y energía a bordo limitados además el entorno es cambiante debido a las condiciones de iluminación débiles y cambiantes Tchernykh et al. (2006). En el trabajo de Shen et al. (2023) se emplea el método de flujo óptico para la conducción autónoma, donde principalmente se hace uso de cámaras de ojo de pez (fisheye camera) para la detección de campos cercanos, por lo que se propuso un algoritmo basado en redes neuronales para la estimación del desplazamiento. Otro campo de aplicación es la video-vigilancia y el análisis de comportamientos, donde surge la necesidad de estimar la velocidad y la dirección del movimiento para analizar comportamientos humanos en grupo como correr, encontrarse, luchar o caerse, por lo que para obtener la magnitud y dirección del movimiento se emplea el flujo óptico Kajo et al. (2015).

3.1 Métodos para el cálculo del flujo óptico

Los algoritmos que calculan el desplazamiento considerando todos los píxeles de la imagen se denominan algoritmos de flujo óptico denso mientras que los algoritmos que estiman el desplazamiento para un número seleccionado de píxeles de la imagen se denominan flujo óptico disperso. Un ejemplo de los algoritmos de flujo óptico denso, es el método de Horn-Schunck que calcula el desplazamiento en cada píxel de la imagen usando restricciones globales, este método evita la extracción de características pero son menos resistentes al ruido. Los algoritmos de flujo óptico disperso, como el de Lucas-Kanade, asumen constantes locales (Nourani-Vatani et al., 2012).

El método de flujo óptico ha sido empleado para estimar el desplazamiento en aglomeraciones de personas, ya que se tiene interés en identificar el comportamiento de multitudes y para esto es necesario estimar la velocidad y dirección del movimiento de los elementos. Se han presentado los métodos basados en el censo del flujo óptico, el método de Black y Anadam, el método de Brox y el método de Zach. Aunque los métodos tratan de afrontar los retos causados por los cambios de iluminación y las discontinuidades en el movimiento (Kajo et al., 2015).

Una mejora al método de Lucas-Kanade es el método de Shi-Tomasi, el cual se basa en la detección de esquinas como puntos característicos de los objetos en movimiento para proporcionar información sobre el movimiento minimizando el cálculo de los puntos (Kajo et al., 2015).

3.1.1 Método de Horn-Schunck

Este algoritmo fundamental, propuesto por Horn and Schunck (1981), donde se emplea una minimización en la función de costo, que combina la optimización de la restricción de flujo óptico con una restricción en la suavidad del campo de vectores de movimiento, este proceso se muestra en la ecuación 3.1. El primer término es el error cuadrático medio de la restricción del movimiento, el segundo término hace referencia a la regularización, el cual permite que el gradiente del campo del vector de movimiento tome valores pequeños (Pesquet-Popescu et al., 2013).

$$J_{HS}(V) = \iint_R \left[u \frac{\partial f}{\partial x} + v \frac{\partial f}{\partial y} + \frac{\partial f}{\partial t} \right]^2 dx dy + \lambda \iint_R [\|\nabla u\|^2 + \|\nabla v\|^2] dx dy \quad (3.1)$$

Después de la solución mediante la minimización de la función de costo de la ecuación 3.1, se llega a las ecuaciones 3.2 y 3.3, descrita para los dos componentes del campo vectorial de movimiento.

$$\lambda \nabla^2 u = \frac{\partial f}{\partial x} \left[u \frac{\partial f}{\partial x} + v \frac{\partial f}{\partial y} + \frac{\partial f}{\partial t} \right] \quad (3.2)$$

$$\lambda \nabla^2 v = \frac{\partial f}{\partial y} \left[u \frac{\partial f}{\partial x} + v \frac{\partial f}{\partial y} + \frac{\partial f}{\partial t} \right] \quad (3.3)$$

3.1.2 Método de Lucas-Kanade

El método de Lucas-Kanade es una técnica ampliamente utilizada en el campo de la visión por computadora para estimar el flujo óptico, que es el patrón aparente de movimiento de objetos, superficies y bordes en una secuencia visual causada por el movimiento relativo entre un observador (una cámara) y la escena. Desarrollado por Bruce D. Lucas y Takeo Kanade en 1981, este método se basa en suposiciones de constancia y suavidad del flujo óptico (Baker and Matthews, 2004). Este método asume que el flujo óptico es constante en una vecindad pequeña alrededor de cada píxel. Esto significa que todos los píxeles dentro de una pequeña ventana alrededor de un píxel en la imagen tienen el mismo movimiento.

Para entender el método, se considera una imagen $I(x, y, t)$ que cambia con el tiempo t . El término (u, v) describe el desplazamiento del píxel $I(x, y, t)$ en la dirección x y y respectivamente como se muestra en la Ecuación 3.4).

$$I(x, y, t) = I(x + u, y + v, t + 1) \quad (3.4)$$

Aplicando una expansión en serie de Taylor a la intensidad de la imagen, se obtiene la ecuación 3.5.

$$I(x + u, y + v, t + 1) \approx I(x, y, t) + \frac{\partial I}{\partial x}u + \frac{\partial I}{\partial y}v + \frac{\partial I}{\partial t} \quad (3.5)$$

De la constancia de la intensidad se obtiene la ecuación 3.6. Esta ecuación es conocida como la ecuación del flujo óptico.

$$\frac{\partial I}{\partial x}u + \frac{\partial I}{\partial y}v + \frac{\partial I}{\partial t} = 0 \quad (3.6)$$

El método de Lucas-Kanade resuelve la ecuación del flujo óptico para todos los píxeles en una pequeña ventana alrededor de cada punto (x, y) . Así para obtener

la solución de este sistema se pueden utilizar diversos métodos numéricos (Kajo et al., 2015).

3.1.3 Método de Shi-Tomasi

El método de Shi-Tomasi es una técnica utilizada en visión por computadora para detectar puntos de interés o características robustas en una imagen, mejorando el rendimiento del rastreo de puntos a través de secuencias de imágenes. Es una extensión del algoritmo de detección de esquinas de Harris y se introdujo en el artículo Shi and Tomasi (1994).

El algoritmo se basa en la idea de que los puntos de interés, como las esquinas y ciertos tipos de bordes, pueden ser rastreados de manera confiable a través de diferentes frames de un video o diferentes imágenes de una secuencia.

La detección de esquinas de Shi-Tomasi se basa en la respuesta de una ventana de imagen a desplazamientos en varias direcciones. Se considera que una esquina es un punto donde hay un cambio significativo en la intensidad en todas las direcciones. Este método utiliza la matriz de autocorrelación de la imagen para determinar estos cambios.

Dada una imagen $I(x, y)$, se considera una ventana W centrada en cada píxel con posición (x, y) . La variación de intensidad en esta ventana para un desplazamiento (u, v) se puede aproximar mediante la ecuación 3.7, donde M es la matriz de autocorrelación dada por la ecuación 3.8.

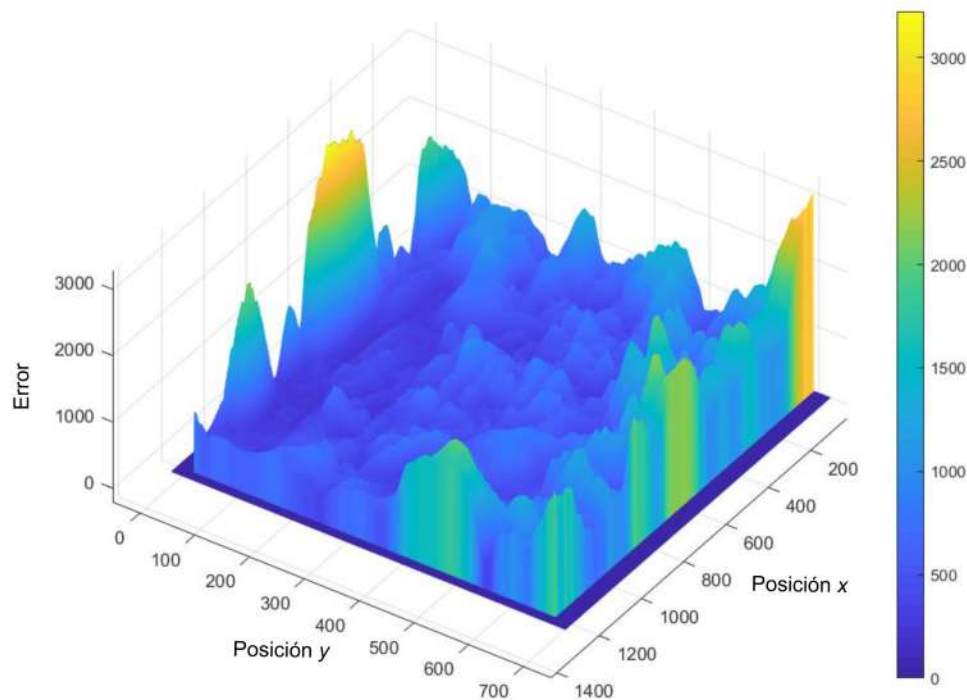
$$E(u, v) \approx \begin{pmatrix} u \\ v \end{pmatrix}^T M \begin{pmatrix} u \\ v \end{pmatrix} \quad (3.7)$$

$$M = \sum_{(x,y) \in W} \begin{pmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{pmatrix} \quad (3.8)$$

En la ecuación 3.8, I_x e I_y representan las derivadas de la intensidad de la imagen en las direcciones x y y , respectivamente. De este modo, se seleccionan los puntos donde se minimicen los valores de la característica (ver Figura 3.1).

Figura 3.1

Ejemplo método de resolución Shi-Tomasi.



3.1.4 Método de basado en algoritmos genéticos

Las metodologías previas, emplean métodos numéricos para la búsqueda de la posición del objeto por lo que tienden a converger en la primera solución que se encuentre, es decir la se detectan zonas muy similares a la de imagen de referencia, y no resulta de la verdadera solución del sistema. Por esta razón Reynosa-Guerrero et al. (2021) propuso el uso de algoritmos genéticos para estimar la región que mejor se ajusta a la solución mediante una búsqueda aleatoria.

A pesar de las mejoras obtenidas en el algoritmo de Reynosa-Guerrero et al. (2021), una de sus limitaciones es que una región objetivo no tenga suficiente

información de textura en términos de intensidades de color, es decir las diferentes intensidades lumínicas de las imágenes pueden afectar el método.

3.2 Aplicaciones de los algoritmos genéticos

Los algoritmos genéticos previamente han sido aplicados a diversas áreas, debido a su capacidad de resolución en comparación a métodos numéricos.

3.2.1 Internet de las cosas

En internet de las cosa (IoT), los algoritmos genéticos han sido aplicado para la correcta clasificación de datos provenientes de dispositivos. Así como la selección inteligente de dispositivos (Alam, 2020).

3.2.2 Señalamiento inteligente del tráfico

El AG puede implementarse en los tráficos inteligentes. Se presentó un esquema de sensores de vehículos inteligentes combinando IoT con los sistemas de autopistas existentes para reducir los tiempos de viaje, las emisiones de contaminantes, el tráfico y las dificultades de desplazamiento. En todas las ciudades y regiones rurales se han construido semáforos en los puntos de cruce de las carreteras (Aljohani and Alam, 2015).

4. METODOLOGÍA

En esta sección de la tesis se describe la metodología empleada para cumplir con los objetivos planteados en la tesis. Se presenta una propuesta de cálculo del desplazamiento empleando codificaciones binarias de una imagen, proveniente de un conjunto de secuencias dentro de un video empleando algoritmos genéticos como método de optimización global. Se describe los pasos empleados para realizar estas codificaciones binarias de las imágenes, y la obtención del conjunto de fotogramas en modelos de ambientes externos. La metodología sigue un enfoque cuantitativo cuasiexperimental basado en Sampieri, 2019 para el seguimiento de objetos en ambientes exteriores con variaciones de iluminación y expuestos a diferentes condiciones ambientales mediante el cálculo del desplazamiento de objetos.

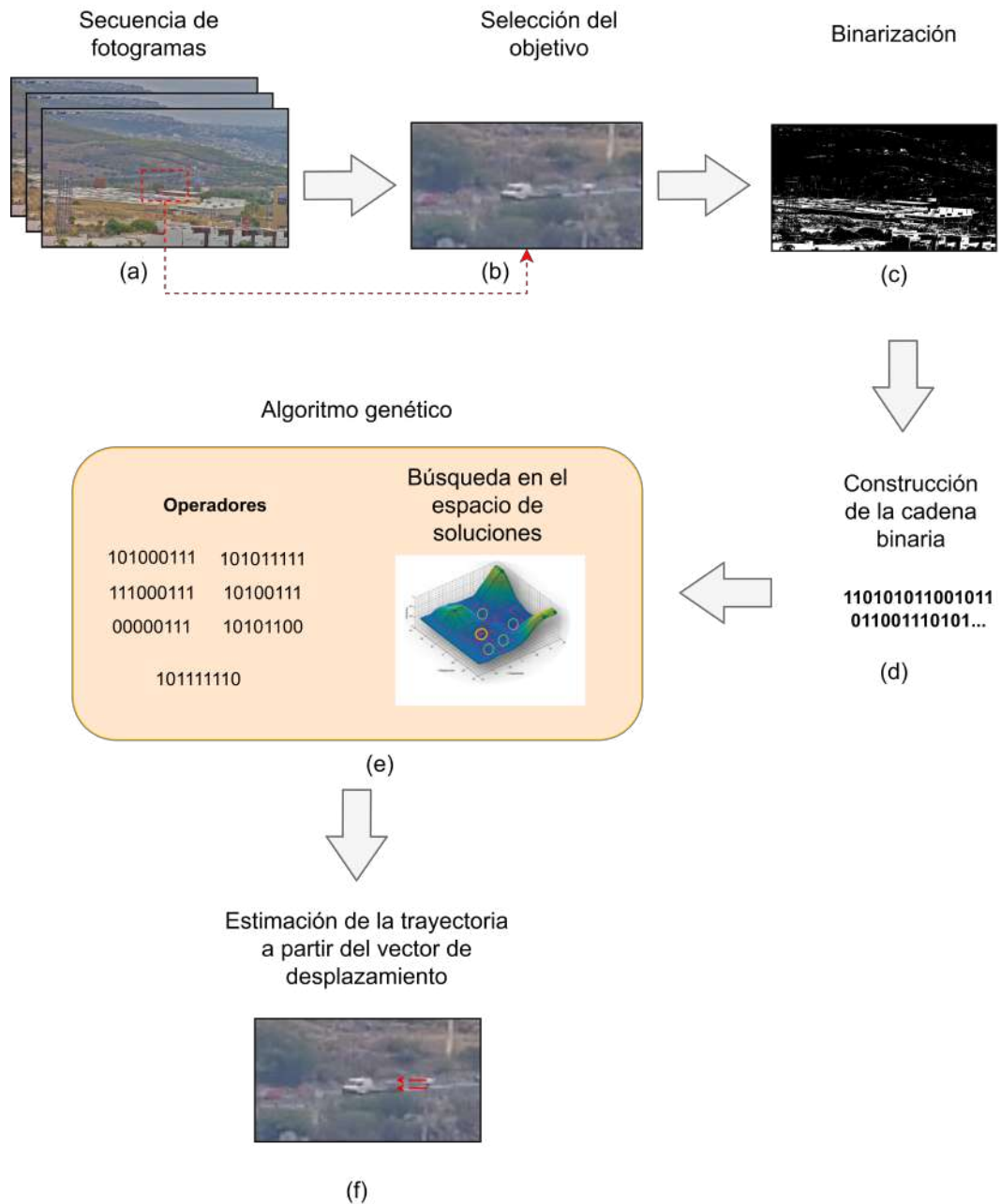
La metodología se ilustra en la Figura 4.1, la cual sigue los siguientes pasos principales:

- Adquisición de secuencias de fotogramas
- Selección del objetivo
- Codificaciones binarias
- Búsqueda por algoritmos genéticos
- Estimación del desplazamiento

4.1 Adquisición de secuencias de fotogramas

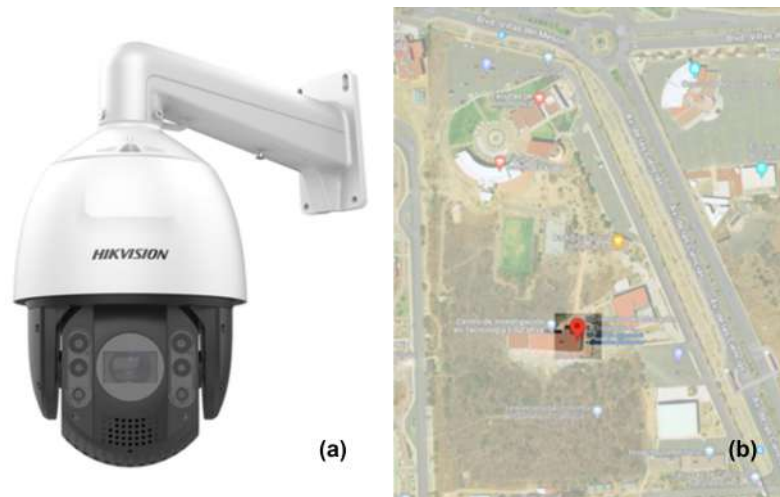
Para adquirir las imágenes que se emplearon para representar el desplazamiento, se dispuso de una cámara de videovigilancia (Figura 4.2a), ubicada en la Facultad de Informática como se muestra en la Figura 4.2b, la cual genera 25 fotogramas por segundo y mantiene una resolución de 1,920 x 1,080 píxeles.

Figura 4.1
Esquema de la metodología.



La cámara proporciona videos con automóviles, los cuales representan un objeto en movimiento mientras que su entorno permanece estático, y se encuentran expuestos a cambios de iluminación y condiciones climáticas variables. Mediante estos videos se realizaron pruebas para la estimación del desplazamiento de los automóviles en dos imágenes consecutivas del video para realizar pruebas del algoritmo propuesto.

Figura 4.2
Instrumentación.

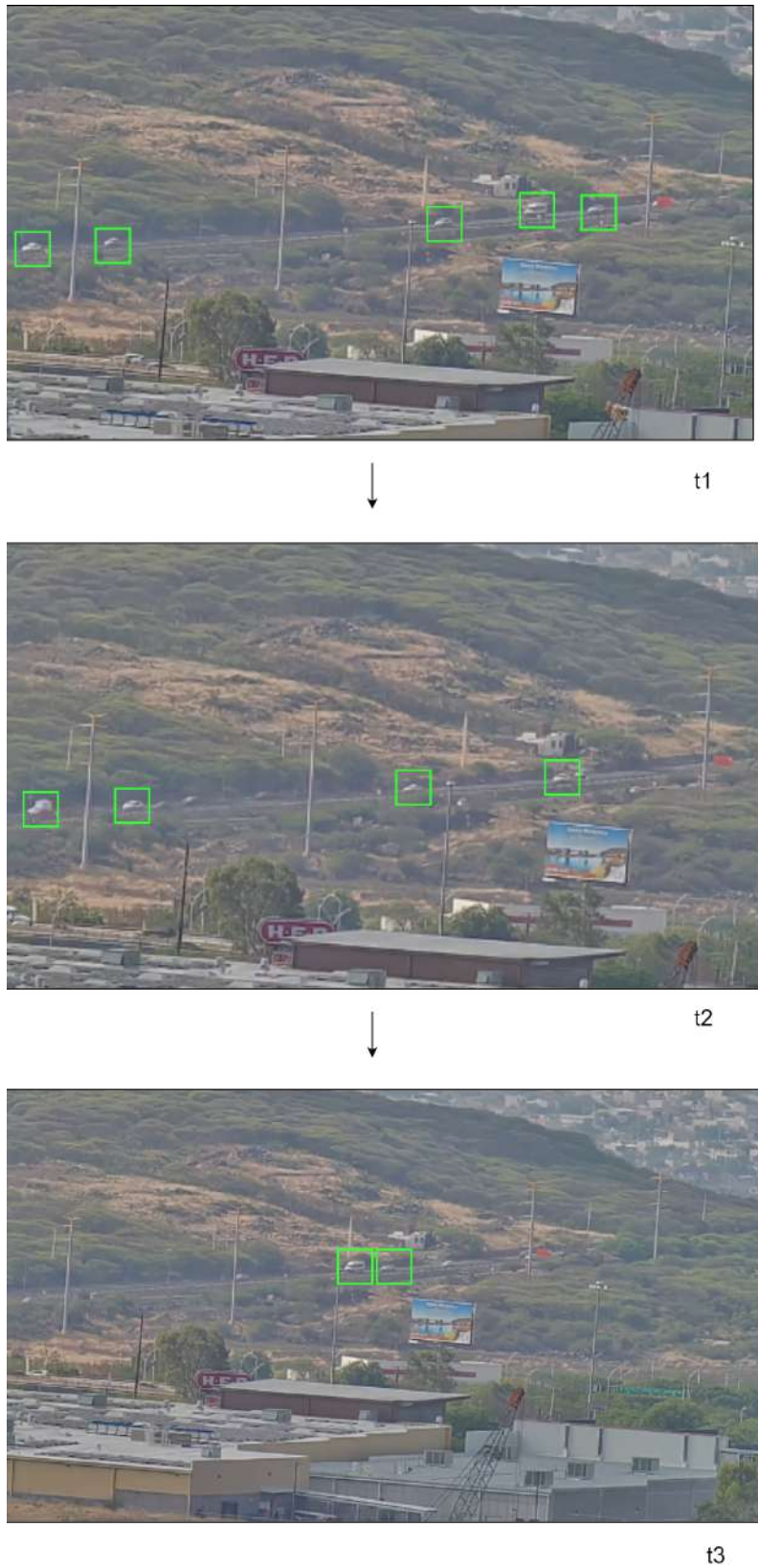


Nota: ilustración a) obtenida de <https://www.pcdigital.com.mx>; ilustración b) Mapa de la ubicación de la cámara.

4.2 Selección del objetivo

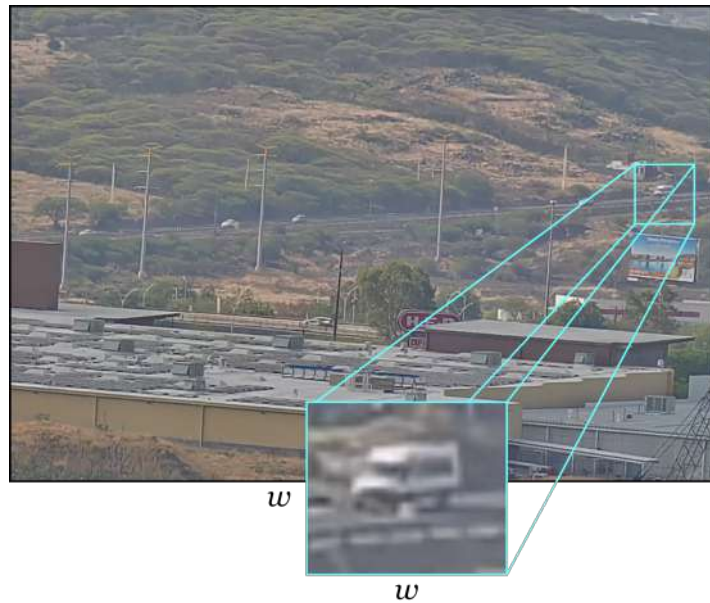
Una vez obtenida una secuencia de fotogramas por medio de un video se segmentan objetos de interés de uno de los fotogramas. Como se ilustra en la Figura 4.3, se selecciona manualmente la posición de un objeto de interés con un tamaño de ventana w , ilustrados en ventanas verdes, al cual se le quiere calcular el desplazamiento (Figura 4.4). Este objeto de interés se encuentra en movimiento, mientras su entorno permanece estático, sin embargo la imagen entre secuencias de fotogramas pueden sufrir cambios de iluminación que no tiene relación con el desplazamiento del objeto. El tamaño de ventana w puede modificarse según el tamaño del objetivo y la representación que se desee tomar del objetivo.

Figura 4.3
Secuencia de fotogramas.



Nota: t1, t2 y t3 corresponden a tiempos consecutivos de una secuencia de imágenes en un video.

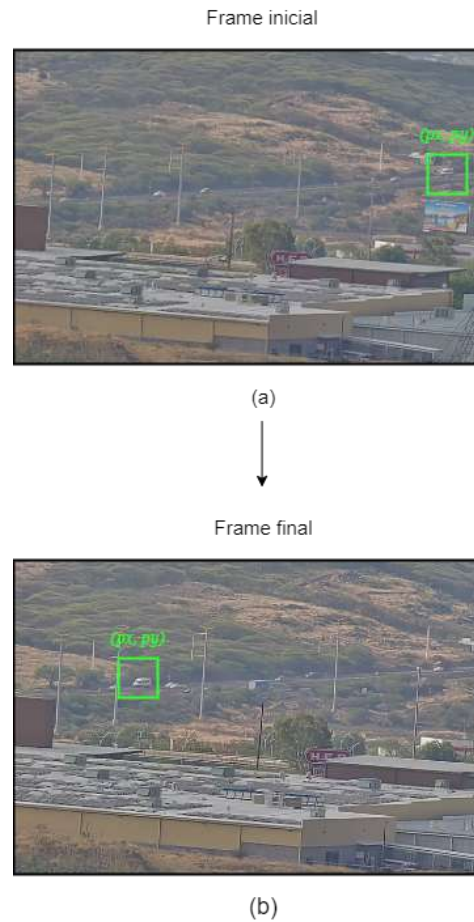
Figura 4.4
Imagen objetivo.



Este objetivo, tiene un tamaño de ventana w de un segmento de la imagen y se encuentra en una posición inicial (x_0, y_0) de la imagen, posteriormente este objetivo se mueve a una posición (x_f, y_f) final, como se ilustra en la Figura 4.5. El cambio de posición del píxel es lo que genera un vector de desplazamiento. Al estimar el desplazamiento, la posición final es desconocida, por lo que el algoritmo de búsqueda, en donde la presente metodología es el algoritmo genético, será el cual ejecute la tarea de encontrar la posición final del objetivo.

Figura 4.5

Imagen objetivo en posición final e inicial.



4.3 Codificaciones binarias

Binarización

Con el objetivo de afrontar las problemáticas obtenidas por los cambios de iluminación, en la metodología se propone un esquema de binarización de la imagen objetivo, es decir la imagen que contiene al objetivo y la imagen de referencia en la cual se desea encontrar el objetivo. Para esto se selecciona un punto de referencia (P_x, P_y) de la imagen objetivo y la imagen de referencia. Las cuales posteriormente se preprocesan mediante la regla especificada por el esquema de codificación binario mostrado en la ecuación 4.1. Esta ecuación denota que si el pixel de la imagen es mayor al punto de referencia (P_x, P_y) , el pixel se simboliza con un 1, mientras que

si es menor, el pixel se simboliza con un 0. De este modo, las imagenes se muestran como en la Figura 4.6.

$$\text{binarizacion}(x, y) = \begin{cases} 0 & \text{si } x, y < P_x, P_y \\ 1 & \text{si } x, y \geq P_x, P_y \end{cases} \quad (4.1)$$

Figura 4.6
Imagen binarizada.



Generación de cadenas binarias

La imagen objetivo de tamaño $w \times w$, cuya posición previamente es seleccionada, se binariza según las reglas de codificación especificadas en el apartado anterior, y se ajusta como un vector para obtener una cadena; el proceso se encuentra ilustrado en la Figura 4.7. Una vez se obtiene la imagen objetivo preprocesada en imagen binaria, se generan posiciones aleatorias para ser tomados como puntos de referencia para la construcción de cadenas binarias que serán tomadas como el conjunto de población inicial. De este modo, la imagen de referencia se segmenta en ventanas de tamaño w , el cual corresponde al tamaño de la imagen objetivo. Cada una de estas segmentaciones se simboliza en una cadena discreta de ceros (0's) y unos (1's) empleando un espacio aleatorio. Así se generan k cadenas binarias

tomando puntos aleatorios. Este proceso se muestra esquematizado en la Figura 4.8.

Figura 4.7

Construcción de cadenas binarias.

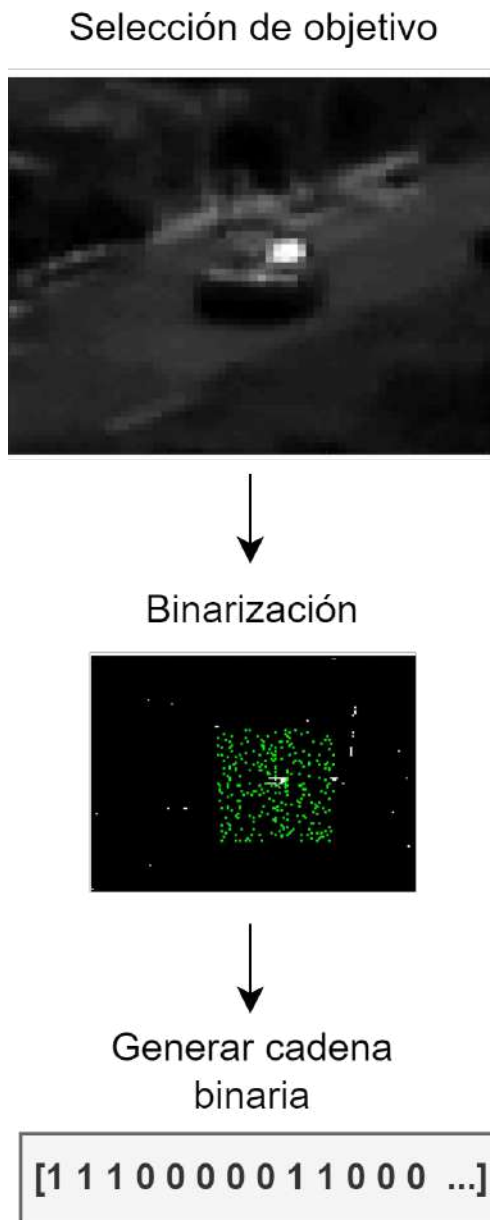
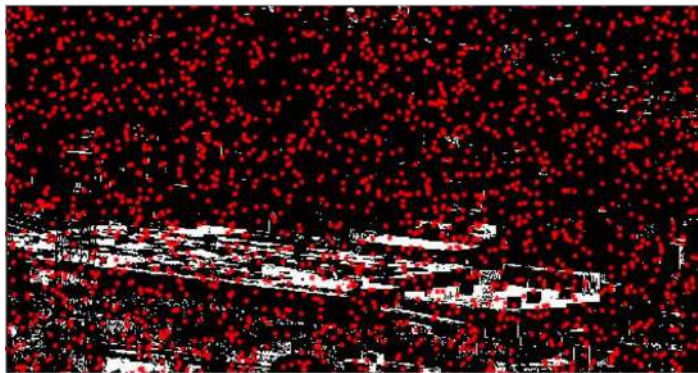


Figura 4.8
Construcción de cadenas binarias de la población

Binarización



Selección puntos aleatorios



Generación de cadenas binarias

```
[0 1 0 1 1 1 1 1 1 0 0 1 1 1...]  
[0 1 1 0 0 1 0 0 1 1 1 0 1...]  
[1 1 1 0 0 0 0 0 1 1 0 0 0 ...]  
[0 0 0 1 1 1 0 0 0 0 0 1 0 ...]
```

4.4 Búsqueda por algoritmo genético

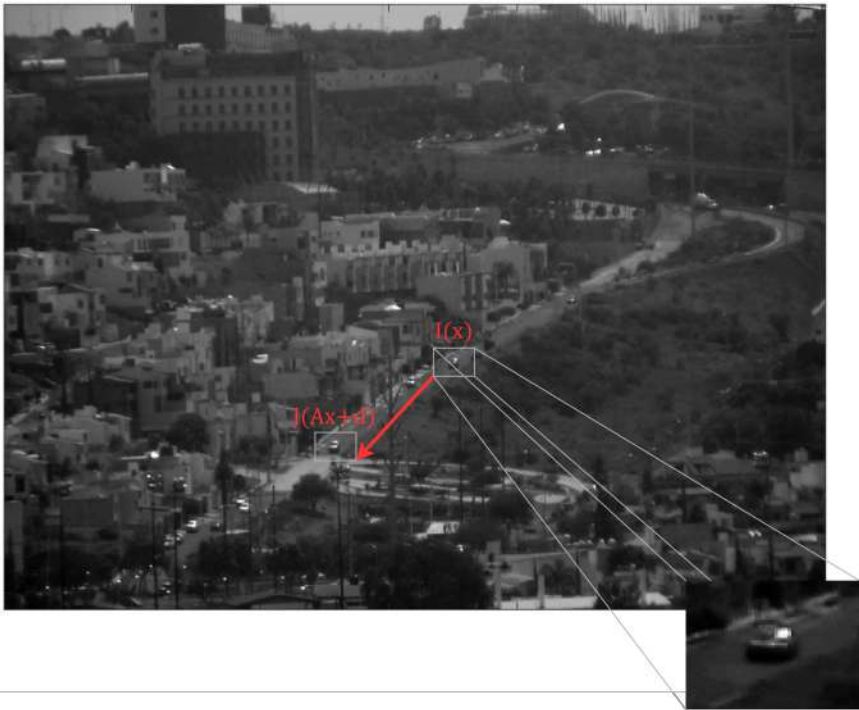
Para la estimación del desplazamiento se empleará lo propuesto por Reynosa-Guerrero et al. (2021) en donde se hace uso de algoritmos genéticos como método de búsqueda dentro del espacio de soluciones para estimar el desplazamiento desde una posición inicial y una posición final como se muestra en la Figura 4.5. Con el aporte propuesto en esta investigación el cual es emplear representaciones binarias de la imagen objetivo y referencia, obtenidas mediante una representación binaria. Esto con la finalidad de que las variaciones de iluminación y que la solución en mínimos locales no afecte en el cálculo del desplazamiento.

4.5 Estimación del desplazamiento

Mediante la solución del algoritmo, se determina la trayectoria del objetivo a través de las posiciones dadas en los resultados del algoritmo de búsqueda mediante la Ecuación 4.2, donde x_0, y_0 representan las posiciones iniciales del objeto y x^* y y^* denotan las posiciones finales encontradas por el algoritmo. Mediante esta ecuación se calcula la trayectoria del objeto en movimiento, como se muestra en la flecha roja de la Figura 4.9

$$v = (x^*, y^*) - (x_0, y_0) \quad (4.2)$$

Figura 4.9
Esquema del desplazamiento.



5. RESULTADOS

En esta sección, se presentan los resultados obtenidos del estudio sobre la estimación del desplazamiento en imágenes utilizando el algoritmo propuesto. Con el propósito de evaluar cómo influye el parámetro k , es decir, el número de cadenas binarias por cada generación de los algoritmos genéticos. Así como el número de generaciones, se realizaron pruebas con diferentes valores de k y número de generaciones. Para el problema de aplicación se empleó una k de 3000.

La cadena objetivo y la población tiene una representación en forma de cadenas binarias y se desea obtener una cadena de la población la cual sea cercana a la cadena objetivo, es decir el número de bits en cada posición sean iguales. De esta forma la función objetivo se implementa como el valor de los bits de la cadena que son diferentes a la cadena objetivo, por ejemplo si en la posición 1 de la cadena objetivo tiene un bit 1 y si en la misma posición de una cadena de la población existe un cero, estos bits desiguales serán penalizados mediante la suma del valor de 1 para todas las posiciones. Se ilustra un ejemplo de este proceso en la Figura 5.1, en rojo se encuentran los puntos diferentes a la cadena objetivo. El valor máximo (Max val), se refiere a la suma de todos los puntos que fueron diferentes a la cadena objetivo. Finalmente se obtuvo el error relativo, que se obtiene en cada generación el cual se muestra en la ecuación 5.1.

$$E_R = \left| \frac{A - V}{V} \right| \times 100\% \quad (5.1)$$

Donde:

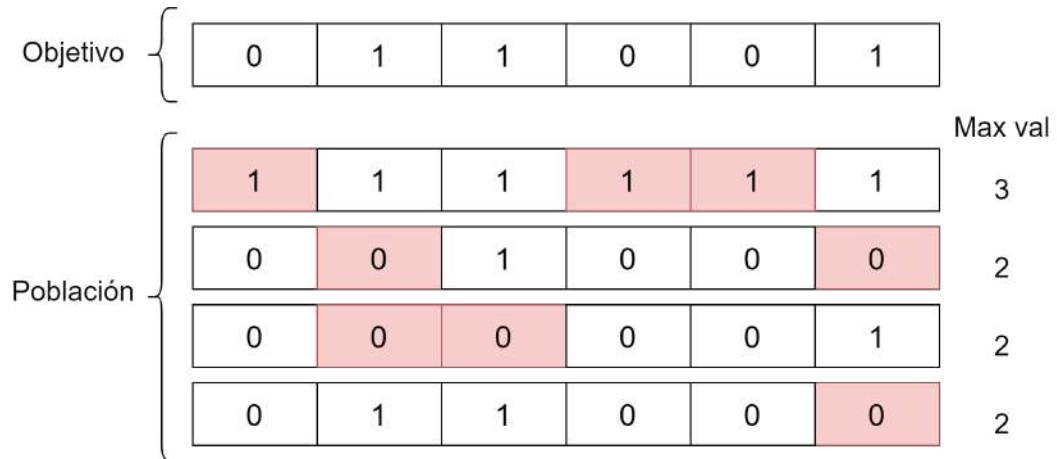
E_R : Es el error relativo porcentual.

A : Es el valor aproximado o estimado.

V : Es el valor verdadero o exacto.

Figura 5.1

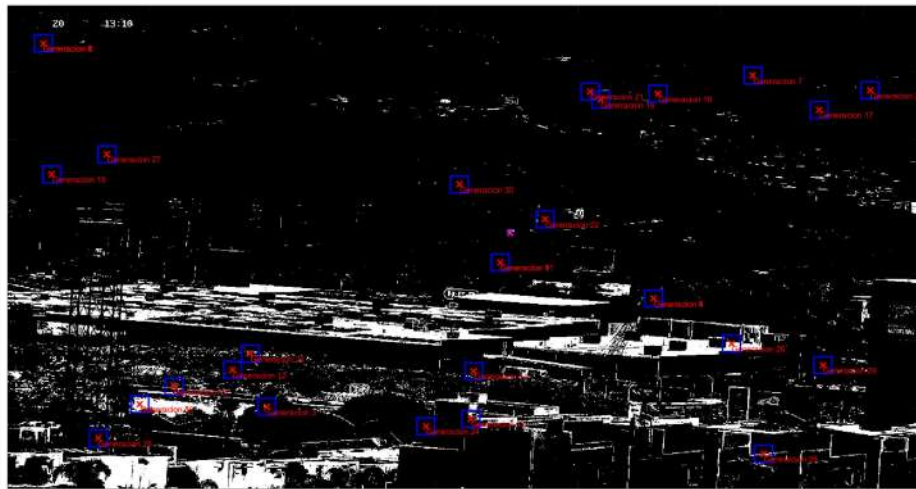
Contenido de bits que difieren de la cadena objetivo.



El proceso general del algoritmo se ilustra en la Figura 5.2, donde las cadenas convergen a cadenas cercanas al el objetivo, que se encuentra marcado con una cruz de color magenta y donde las cruces rojas son las posiciones dadas por el algoritmo genético. Se empieza por una posición aleatoria hasta converger a la cadena del objetivo.

Figura 5.2

Metodología con 30 generaciones.



El objeto seleccionado, empieza en una posición 705 y 339, en x y y respectivamente con un tamaño de ventana w de 100, el cual corresponde a un automóvil en movimiento, esta posición se encuentra marcada con una cruz de color magenta

de la Figura 5.2. Esta posición corresponde a la posición para obtener k , cadenas binarias con n número de bits.

Para los resultados obtenidos se emplearon 10, 50, 100, 500 y 1000 generaciones que se muestran en las Figuras 5.3, 5.4, 5.5, 5.6 y 5.7 respectivamente. En ellas se muestra la disminución del error porcentual por cada generación.

Los resultados de las pruebas sobre el número de generaciones realizadas en el algoritmo genético, se encuentran resumidos en la tabla 5-1. Como se muestra el error disminuyó empleando un número de generaciones de 300, y más de ese valor, error relativo se mantuvo constante, por lo que el número de generaciones óptimas para el problema de aplicación es de 300.

Tabla 5-1

Resultados de experimentos realizados

Número de generaciones	max_val	erro_rel
10	24	6
20	24	6
30	24	6
50	24	6
80	23	6
100	22	5.5
300	20	5
500	20	5
1000	20	5

En la figura 5.8 se muestran los errores relativos porcentuales obtenidos en cada generación, según el color del círculo y los círculos representan las posiciones de cada generación.

Figura 5.3
Resultados 10 generaciones.

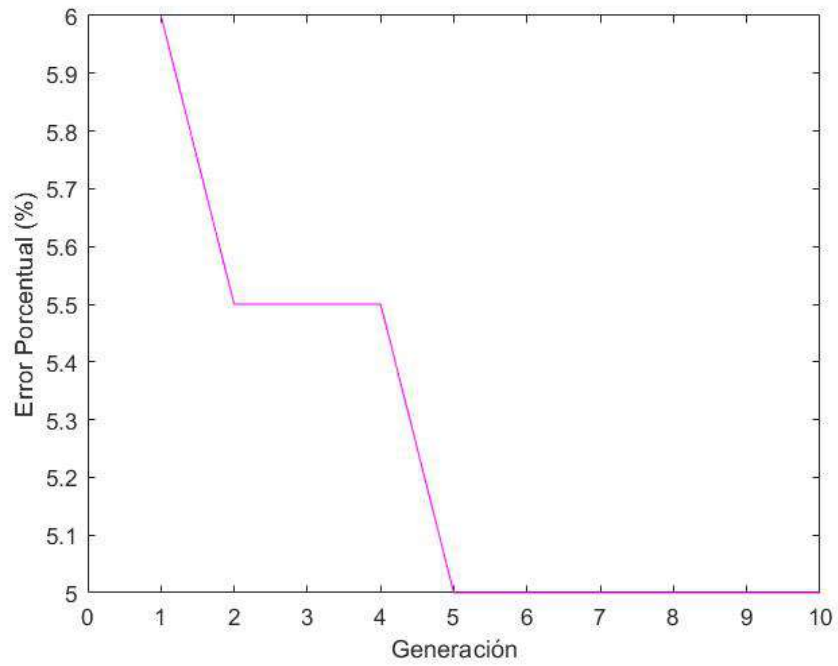


Figura 5.4
Resultados 50 generaciones.

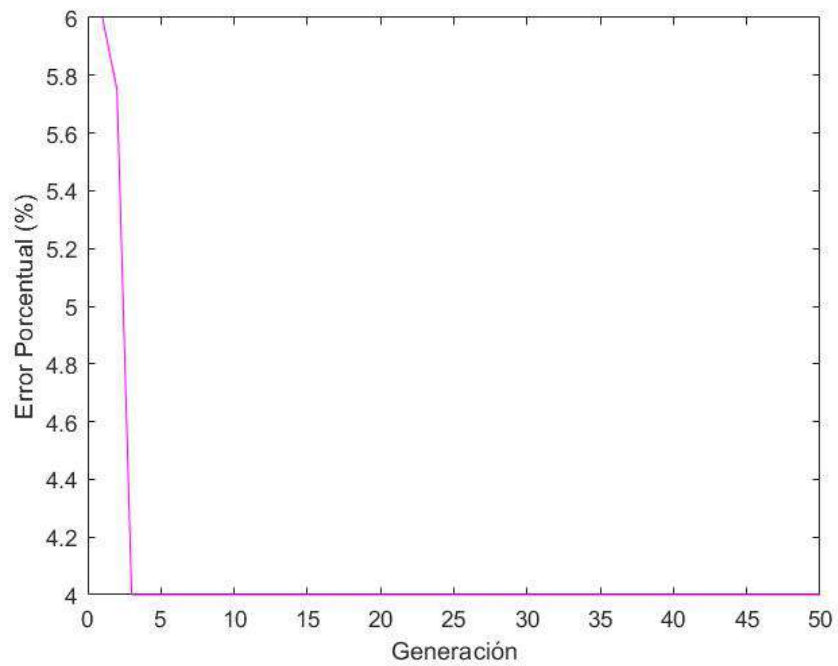


Figura 5.5
Resultados 100 generaciones.

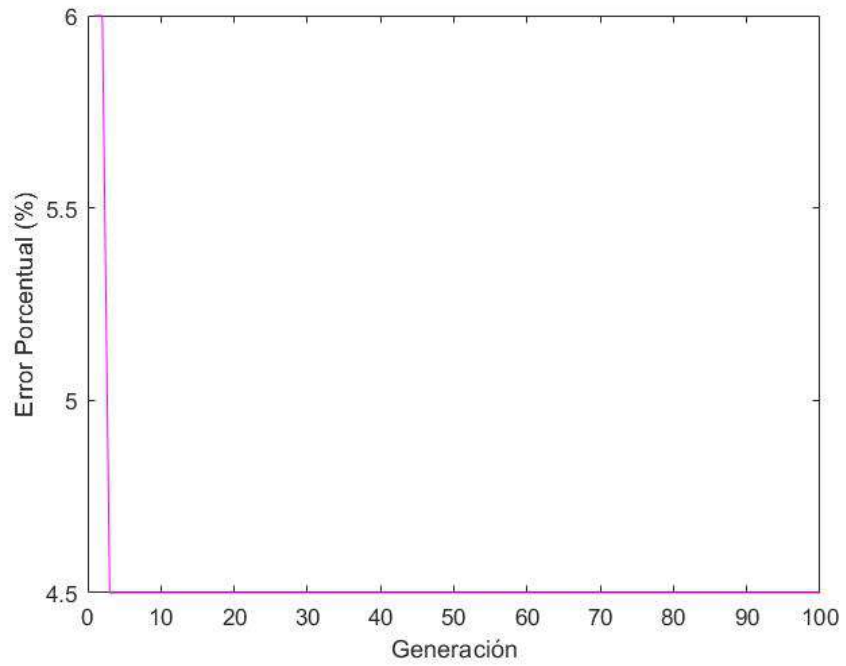


Figura 5.6
Resultados 500 generaciones.

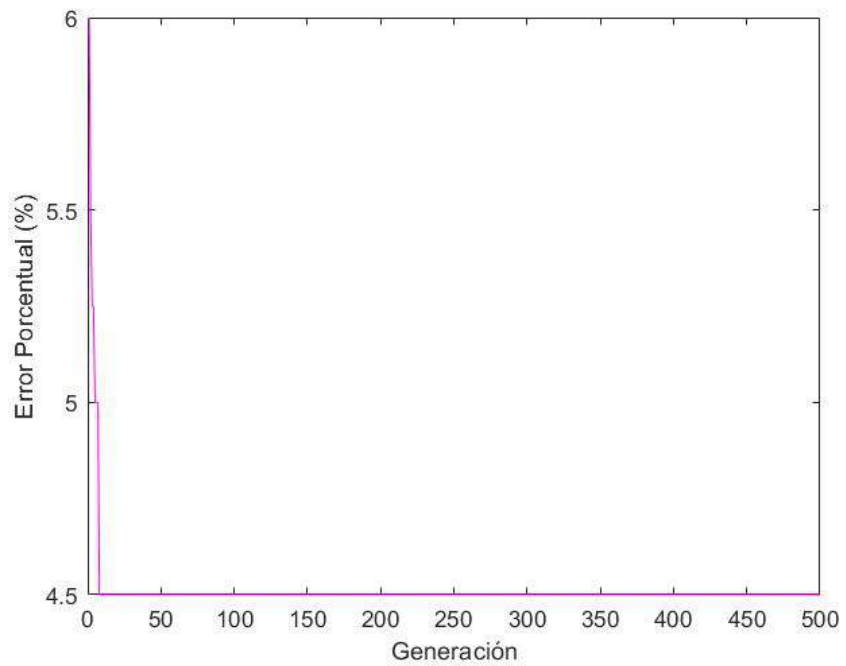


Figura 5.7
Resultados 1000 generaciones.

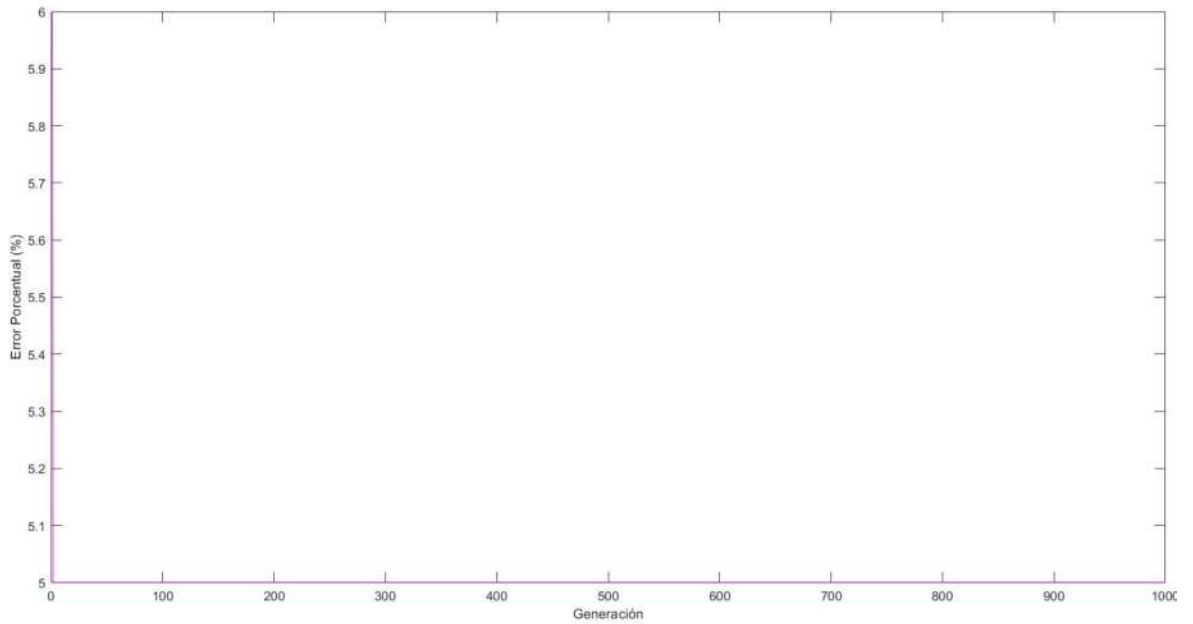


Figura 5.8
Errores obtenidos.



6. CONCLUSIONES Y TRABAJO FUTURO

En este estudio, se ha presentado un algoritmo efectivo para la estimación del desplazamiento en imágenes, basado en el método de flujo óptico y la solución por medio de algoritmos genéticos. Este enfoque ha demostrado ser prometedor al proporcionar una estimación precisa del movimiento en diferentes condiciones de iluminación. Al utilizar las diferentes generaciones del algoritmo y variar el tamaño de posiciones t tamaño de las cadenas, este algoritmo ha demostrado ser robusto y eficiente en la estimación de desplazamientos incluso en presencia de ruido y cambios de iluminación en la escena.

La comparación cuantitativa con otros métodos ha demostrado la mejora del enfoque en una escenas de variación de iluminación. Además, los resultados cuantitativos del error relativo porcentual han ilustrado la capacidad del algoritmo de obtener diferentes posiciones del movimiento empleando una representación simbólica de la imagen mediante cadenas binarias.

En resumen, el algoritmo ofrece una solución eficaz y confiable para la estimación del desplazamiento en imágenes, con amplias aplicaciones potenciales en campos como la visión por computadora, el seguimiento de objetos, la estabilización de video, entre otros.

Como trabajo a futuro, se pueden explorar la extracción y codificación de las características de las imágenes para se usados en la función de aptitud.

7. REFERENCIAS

- Alam, T. (2020). Efficient and secure data transmission approach in cloud-manet-iot integrated framework. *Industrial & Manufacturing Engineering eJournal*.
- Alhijawi, B. and Awajan, A. (2023). Genetic algorithms: theory, genetic operators, solutions, and applications. *Evolutionary Intelligence*, 17:1–12.
- Aljohani, M. and Alam, T. (2015). An algorithm for accessing traffic database using wireless technologies. *2015 IEEE International Conference on Computational Intelligence and Computing Research (ICIC)*, pages 1–4.
- Artizzu, C.-O., Zhang, H., Allibert, G., and Demonceaux, C. (2021). Omniflownet: a perspective neural network adaptation for optical flow estimation in omnidirectional images. *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 2657–2662.
- Baeck, T., Fogel, D. B., and Michalewicz, Z. (2000). Evolution-ary computation 1: Basic algorithms and operators.
- Baker, S. and Matthews, I. (2004). Lucas-kanade 20 years on: A unifying framework. *International Journal of Computer Vision*, 56:221–255.
- Balaji, S. R. and Karthikeyan, S. (2017). A survey on moving object tracking using image processing. In *2017 11th International Conference on Intelligent Systems and Control (ISCO)*, pages 469–474.
- Carlson, W. E. (1991). A survey of computer graphics image encoding and storage formats. *SIGGRAPH Comput. Graph.*, 25(2):67–75.
- Chen, J., Lai, J., Cai, Z., Xie, X., and Pan, Z. (2021). Optical flow estimation based on the frequency-domain regularization. *IEEE Transactions on Circuits and Systems for Video Technology*, 31:217–230.

- Choi, S., Cha, S.-H., and Tappert, C. (2009). A survey of binary similarity and distance measures. *J. Syst. Cybern. Inf.*, 8.
- Cicirello, V. A. (2024). Evolutionary computation: Theories, techniques, and applications. *Applied Sciences*, 14(6).
- Davies, E. (2018). Chapter 20 - motion. In Davies, E., editor, *Computer Vision (Fifth Edition)*, pages 611–628. Academic Press, fifth edition edition.
- Duda, R. O., Hart, P. E., and Stork, D. G. (2000). *Pattern Classification (2nd Edition)*. Wiley-Interscience, 2 edition.
- Fortun, D., Bouthemy, P., and Kervrann, C. (2015). Optical flow modeling and computation: A survey. *Computer Vision and Image Understanding*, 134:1–21. Image Understanding for Real-world Distributed Video Networks.
- Goshtasby, A. A. (2012). *Image Registration*. Springer London.
- Haq, I. U. and Caballero, J. (2021). A survey of binary code similarity. *ACM Comput. Surv.*, 54(3).
- Heaton, J. (2014). *Artificial Intelligence for Humans: Nature-inspired algorithms*. Artificial Intelligence for Humans. Heaton Research, Incorporated.
- Holland, J. H. (1992). Genetic algorithms. *Scientific American*, 267(1):66–73.
- Horn, B. K. and Schunck, B. G. (1981). Determining optical flow. *Artificial Intelligence*, 17(1):185–203.
- Hu, L., Zhao, R., Ding, Z., Ma, L., Shi, B., Xiong, R., and Huang, T. (2021). Optical flow estimation for spiking camera. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 17823–17832.
- Ismagilova, E., Hughes, L., Dwivedi, Y. K., and Raman, K. R. (2019). Smart cities: Advances in research—an information systems perspective. *International Journal of Information Management*, 47:88–100.

- Jusman, Y., Hinggis, L., Wiyagi, R. O., Isa, N. A. M., and Mujaahid, F. (2020). Comparison of background subtraction and frame differencing methods for indoor moving object detection. In *2020 1st International Conference on Information Technology, Advanced Mechanical and Electrical Engineering (ICITAMEE)*, pages 214–219.
- Kajo, I., Malik, A. S., and Kamel, N. (2015). Motion estimation of crowd flow using optical flow techniques: A review. In *2015 9th International Conference on Signal Processing and Communication Systems (ICSPCS)*, pages 1–9.
- Katoch, S., Chauhan, S. S., and Kumar, V. (2021). A review on genetic algorithm: past, present, and future. *Multimedia tools and applications*, 80:8091–8126.
- Kleyko, D. (2016). *Pattern Recognition with Vector Symbolic Architectures*. PhD thesis, Dept. of Computer Science and Electrical Engineering Luleå University of Technology Luleå, Sweden.
- Kulchandani, J. S. and Dangarwala, K. J. (2015). Moving object detection: Review of recent research trends. In *2015 International Conference on Pervasive Computing (ICPC)*, pages 1–5.
- Kumar, C. K. and Rawal, K. (2022). A brief study on object detection and tracking. *Journal of Physics: Conference Series*, 2327(1):012012.
- Manchanda, S. and Sharma, S. (2016). Analysis of computer vision based techniques for motion detection. In *2016 6th International Conference - Cloud System and Big Data Engineering (Confluence)*, pages 445–450.
- Mirjalili, S., Song Dong, J., Sadiq, A. S., and Faris, H. (2020). *Genetic Algorithm: Theory, Literature Review, and Application in Image Reconstruction*, pages 69–85. Springer International Publishing, Cham.
- Mittal, S. and Nag, S. (2019). A survey of encoding techniques for reducing data-movement energy. *Journal of Systems Architecture*, 97:373–396.
- Nag, S. (2017). Image registration techniques: A survey.

- Nourani-Vatani, N., Borges, P. V. K., and Roberts, J. M. (2012). A study of feature extraction algorithms for optical flow tracking. In *IEEE International Conference on Robotics and Automation*.
- Pan, J.-S., Hu, P., Snášel, V., and Chu, S.-C. (2022). A survey on binary metaheuristic algorithms and their engineering applications. *Artificial Intelligence Review*, 56(7):6101–6167.
- Pan, L., Liu, M., and Hartley, R. I. (2020). Single image optical flow estimation with an event camera. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1669–1678.
- Pesquet-Popescu, B., Cagnazzo, M., and Dufaux, F. (2013). Motion estimation techniques.
- Pérez-Rúa, J.-M., Crivelli, T., and Pérez, P. (2016). Object-guided motion estimation. *Computer Vision and Image Understanding*, 153:88–99. Special issue on Visual Tracking.
- Reynosa-Guerrero, J., Garcia-Huerta, J.-M., Vazquez-Cervantes, A., Reyes-Santos, E., Perez-Ramos, J.-L., and Jimenez-Hernandez, H. (2021). Estimation of disparity maps through an evolutionary algorithm and global image features as descriptors. *Expert Systems with Applications*, 165:113900.
- Santhi, D. K. and Vinodhini, D. V. (2021). Study on selection methods of parents and crossover in genetic algorithm. *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*.
- Shah, S. and Xuezhi, X. (2021). Traditional and modern strategies for optical flow: an investigation. *SN Applied Sciences*, 3.
- Shen, S., Kerofsky, L., and Yogamani, S. K. (2023). Optical flow for autonomous driving: Applications, challenges and improvements. *ArXiv*, abs/2301.04422.
- Shi, J. and Tomasi, C. (1994). Good features to track. *1994 Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pages 593–600.

- Tchernykh, V., Beck, M., and Janschek, K. (2006). Optical flow navigation for an outdoor uav using a wide angle mono camera and dem matching. *IFAC Proceedings Volumes*, 39(16):590–595. 4th IFAC Symposium on Mechatronic Systems.
- Yan, Z. and Xiang, X. (2016). Scene flow estimation: A survey. *ArXiv*, abs/1612.02590.
- Zhai, M., Xiang, X., Lv, N., and Kong, X. (2021). Optical flow and scene flow estimation: A survey. *Pattern Recognition*, 114:107861.
- Zitová, B. and Flusser, J. (2003). Image registration methods: a survey. *Image and Vision Computing*, 21(11):977–1000.