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

Procesamiento y Análisis de Imágenes en Smartphone mediante una mejora del algoritmo SIFT

Opción de titulación
Tesis

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

Presenta:
Raúl David Palma Olvera

Dirigido por:
Dr. Jesús Carlos Pedraza Ortega

Dr. Jesús Carlos Pedraza Ortega Presidente	Firma
Dr. Juan Manuel Ramos Arreguín Secretario	Firma
Dr. Saúl Tovar Arriaga Vocal	Firma
Dr. Hugo Jiménez Hernández Suplente	Firma
Dr. Jorge Alberto Soto Cajiga Suplente	Firma

M.C. Ruth Angélica Rico Hernández
Directora de la Facultad

Dra. Ma. Guadalupe Flavia Loarca Piña
Director de Investigación y Posgrado

Centro Universitario
Querétaro, Qro.
Abril, 2015.
México.

REPOSITORIO INSTITUCIONAL DGBSDI-UAQ

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.

RESUMEN

En este trabajo se propone la mejora del desempeño del algoritmo de extracción de características SIFT (Scale Invariant Feature Transform) para el reconocimiento de edificios en condiciones de iluminación no controladas al aire libre. Para ello se propone establecer una comparación entre la cantidad de características extraídas con el algoritmo SIFT original y la propuesta planteada que añade una etapa de preprocesamiento basada en realce de histogramas con ayuda del algoritmo CLAHE (Contrast Limited Adaptive Histogram Equalization). Dicho algoritmo será utilizado para mejorar las imágenes tomadas con variaciones de iluminación drásticas, es decir, con una cantidad de iluminación precaria y también con una gran cantidad de iluminación.

Este proceso mejorara la extracción de características en dichas condiciones lo cual favorecerá al algoritmos SIFT para el reconocimiento y emparejamiento de las estructuras a encontrar en la escena. Se probara esta metodología propuesta con imágenes tomadas a distintas horas del día lo cual nos dará a conocer en qué momento es más viable la extracción de características o si el comportamiento es el mismo aun con los cambios de iluminación.

Una vez que se halla comprobado la metodología propuesta se hará una implementación del método propuesto en un dispositivo basado en el sistema operativo Android logrando una mayor movilidad en el sistema para la realización de las pruebas al aire libre.

(**Palabras clave**: Algoritmo SIFT, Algoritmo CLAHE, Ecualización de Histogramas, Extracción de Características, Reconocimiento de Edificios)

SUMMARY

In this work a performance improvement to the feature extraction algorithm SIFT (Scale Invariant Feature Transform) is proposed for the recognition of buildings in uncontrolled lighting conditions, at outdoors. For it is proposed to establish a comparison between the amount of features extrated with the original SIFT algorithm and the proposed method that adds a preprocessing stage based on enhanced histograms using the CLAHE (Contrast Limited Adaptive Histogram Equalization) algorithm. This algorithm will be used to improve the images taken with drastic ligthing variations, ie, with precarious lighting conditions and also with a great amount of lighting.

This process will improve the feature extraction in these lighting conditions which will favor the SIFT algorithm for the recognitions and matching of buildings at the scene. The proposed methodology will be tested with images taken at different times of days which will give us to know at what point is most feasible the feature extraction or if the behavior is the same even with changes in the lighting conditions.

Once the proposed methodology is verified, an implementation of this will be developed on a mobil device based on the Android operative system to achieve greater mobility in the system for testing outdoors.

(**Key words**: SIFT Algorithm, CLAHE Algorithm, Histogram Equalization, Feature Extraction, Building Recognition)

A mi familia, amigos y seres queridos que me han acompañado a lo largo la vida.

AGRADECIMIENTOS

Agradezco al Consejo de Ciencia y Tecnología (CONACYT) y a la Universidad Autónoma de Querétaro y al Centro de Ingeniería y Desarrollo Industrial (CIDESI), por brindar el apoyo y las facilidades para desarrollar este proyecto.

Al núcleo académico a cargo del programa de posgrados de esta facultad de informática y principalmente al Dr. Jesús Carlos Pedraza Ortega por su guía en este trabajo y durante toda la etapa de este posgrado, al Dr. Juan Manuel Ramos Arreguín el cual me brindo apoyo y orientación en múltiples ocasiones y considerándome en actividades que me ayudaron a desarrollarme en múltiples ámbitos escolares.

También agradezco al Dr. Hugo Jiménez Hernández por sus valiosas observaciones y enseñanzas las cuales hicieron posible un análisis más detallado del proyecto y me ayudaron dar un mejor enfoque al trabajo. A Dr. Jorge Alberto Soto Cajiga por los consejos, las enseñanzas y el apoyo brindado durante la estancia realizada en CIDESI.

Agradezco infinitamente a mi familia que me ha apoyado en todas mis decisiones a lo largo de mi vida y comprendiendo que estas me alejarían temporalmente de ellos, pero que siempre los llevaría en mi mente.

De la misma manera agradezco todo el apoyo que me brindo mi novia y compañera de vida Elizabeth Martínez que me ha acompañado en los buenos y malos momentos de este proceso, y me ha apoyado y orientado en los momentos que creía perder el enfoque.

Y por último, pero no menos importante a mis amigos, los cuales en ocasiones tuve que dejar de lado para poder realizar las actividades demandantes que implican el realizar una maestría, pero siempre comprendieron y estuvieron conmigo.

Gracias.

TABLA DE CONTENIDOS

1. INTRODUCCIÓN 11

1.1 DEFINICION DEL PROYECTO DE INVESTIGACION. 11

1.2 JUSTIFICACIÓN 11

1.3 OBJETIVOS DE LA INVESTIGACION 12

1.4 ALCANCES Y LIMITACIONES 13

1.5 ESTRUCTURA DE LA TESIS. 14

2. ANTECEDENTES,TRABAJO RELACIONADO Y FUNDAMENTACIÓN TEÓRICA 16

2.1 PROCESAMIENTO DE IMÁGENES 16

2.2 ALGORITMOS DE VISIÓN POR COMPUTADORA 16

2.3 DETECCION DE CARACTERISTICAS 19

2.4 DETECTOR DE ESQUINAS, BORDES Y TEXTURA ..... **¡ERROR! MARCADOR NO DEFINIDO.**19

2.5 ALGORITMOS BASADOS EN EL EMPAREJAMIENTO DE CARACTERISTICAS: DETECTORES DE ESQUINAS Y BORDES 20

2.5.1 DETECTOR DE ESQUINAS FAST: FEATURES FROM ACCELERATED SEGMENT TEST 22

2.6 ALGORITMOS BASADOS EN TEXTURA 23

2.6.1 LA PROPUESTA DE DAVID LOWE 24

2.6.2 LA PROPUESTA DE MIKOLAJCZYK Y SCHIMD 25

2.7 ALGORITMO SIFT (SCALE INVARIANT FEATURE TRANSFORM) 25

2.7.1 CONSTRUCCION DE PIRAMIDE SCALE-SPACE Y DETECCION DE EXTREMOS 26

2.7.2 LOCALIZACION DE PUNTOS CLAVE 30

2.7.3 ASIGNACION DE ORIENTACION 31

2.7.4 DESCRIPTOR DE PUNTOS CLAVE 33

3. METODOLOGÍA 34

3.1 IMÁGENES UTILIZADAS EN EL ESTUDIO 35

3.2 SOFTWARE PROPUESTO PARA LA INVESTIGACION 36

3.3 OBTENCION DEL HISTOGRAMA DE LAS IMAGENES 36

3.4 ECUALIZACION DE HISTOGRAMA. 39

3.5 ALGORITMO CLAHE 41

3.6 EXTRACCION DE CARACTERISTICAS MEDIANTE EL ALGORITMO SIFT 43

3.7 RECONOCIMIENTO DE LAS IMAGENES 45

4. RESULTADOS Y ANÁLISIS 46

5. CONCLUSIONES Y TRABAJO FUTURO 47

6. REFERENCIAS 48

7. APÉNDICE 50

ÍNDICE DE TABLAS

3-1 Cantidad de características extraídas en cada una de las imágenes patrón....44

3-2 Cantidad de características emparejadas con el mejor y el peor de los casos de las imágenes de prueba a lo largo del día de la base de datos. 46

ÍNDICE DE IMAGENES

2-1 Detección de bordes con el detector Harris. 21

2-2 Deteccion de Esquinas con el algoritmo FAST. 23

2-3 Imagen en diferentes escalas 27

2-4 Por cada octava en las diferentes escalas se aplica una diferencia de Gaussiana. 28

2-5 Diferencia de Gaussiana en imágenes reales 29

2-6 Cada uno de los pixeles de la imagen es comparado con sus pixeles vecinos en la piramide de diferencia de Gaussianas. 30

2-7 Representacion de las caracteristicas extraidas con sus respectivas direcciones.. 32

2-8 Descriptor obtenido por el algoritmo SIFT 33

3-1 Metodologia propuesta 34

3-2 Parte representativa de la base de datos de imágenes obtenida de uno de los edificios de la Facultad de Informática de la UAQ en las que se puede apreciar el cambio de iluminación a lo largo del día. 35

3-3 Imagen de día en escala de grises tomada como patrón y su histograma obtenido de la imagen sin ningun tipo de realce 37

3-4 Imagen de noche en escala de grises tomada como patrón (arriba), histograma obtenido de la imagen original sin ningún tipo de realce(abajo). 38

3-5 Imagen tomada en el día con un procesamiento de ecualización tradicional y su histograma obtenido de la imagen. 39

3-6 Imagen tomada en la noche con un procesamiento de ecualización tradicional y su histograma obtenido de la imagen 40

3-7 Imagen tomada en el día con un realce de histogramas mediante el algoritmo CLAHE (Arriba) y su histograma (Abajo).. 42

3-8 Imagen tomada en la noche con un realce de histogramas mediante el algoritmo CLAHE (Arriba) y su histograma (Abajo).. 43

3-9 Representación de las características extraídas en una imagen de día (Arriba), una imagen por la tarde (Izquierda) y una imagen de noche (derecha) del mismo edificio.....44

3-10 Edificio reconocido por el algoritmo durante el día con la mejora propuesta del algoritmos SIFT.. 45

4-1 Implementación del algoritmo SIFT en Android y Reconocimiento del Edificio.. 47

1. INTRODUCCIÓN

1.1 **Definición del Proyecto de Investigación**

En este proyecto se hace un estudio comparativo del desempeño del algoritmo SIFT (Scale-Invariant Feature Transform) para la detección de estructuras al aire libre, para ello se utilizaran una serie de métodos para el preprocesamiento de las imágenes utilizadas, comparando cada uno de los resultados obtenidos en un ambiente no controlado.

1.2 **Justificación**

La extracción de características y reconocimiento de patrones son dos tareas muy importantes en el reconocimiento de estructuras y objetos, sus aplicaciones van creciendo día a día en diversos campos y aplicaciones, como son clasificación de video (Jinxia), reconocimiento de objetos (Lowe, 2004), Seguimiento de objetos(Langlotz, 2011), clasificación de imágenes (Ayers, 2007), captura de movimiento , reconstrucción de fotos panorámicas , vigilancia móvil , autentificación fácial , entre otras. Se ha seleccionado el algoritmo SIFT (Scale-Invariant Feature Extracción) el cual extrae características de una imagen de manera estable, siendo prácticamente invariable a translación, rotación, escala, iluminación lineal y oclusión. El uso extendido del algoritmo SIFT es atribuido tanto a la localización invariante exitosa de los puntos de interés en posición y escala, este usa un detector basado en diferencias de Gaussianas (DoG) (Lowe, 1999), y un descriptor de puntos clave de 128 elementos (Lowe, 2004).

Este método este útil para hacer un emparejamiento exitoso dependiendo de los tipos de imágenes que van a ser emparejados y las variaciones entre dichas imágenes, algunos de los parámetros son los siguientes:

Escala: Al menos dos elementos de los sets de imágenes tienen diferentes escalas

Oclusión: Es el concepto de que dos objetos que están separados espacialmente en un plano 3d que interfieren uno con el otro en un plano 2D para

una tarea de una sola vista, como es el reconocimiento de objetos, la oclusión es un problema en los algoritmos que no llegan a ser tan robustos.

Orientación: Se refiere a la rotación de una imagen con respecto a otra, un máximo en la orientación de rotación es de 30 grados, es el valor máximo típico para la mayoría de los algoritmos para un emparejamiento sustentable.

Objetos a Emparejar: Si el objeto a identificar es plano, tiene textura o es curvo.

Desordenamiento: Se refiere a la condición del fondo de la imagen, en ocasiones suele provocar resultados poco favorables o le es difícil al algoritmo en entender los límites del objeto de interés cuando el fondo está muy desordenado.

Iluminación: Uno de los principales problemas para un emparejamiento preciso de imágenes.

La mayoría de los algoritmos se desempeñan bien en presencia de los factores antes mencionados, pero muy difícilmente alguno de los algoritmos verdaderamente a demostrado total invariancia a esos parámetros.

Se pretende con el método que se va a proponer el mejorar la extracción de características y el emparejamiento del algoritmo SIFT con imágenes con algunos de los factores antes mencionados, como son iluminación, orientación, escala y oclusión, en un ambiente de al aire libre.

1.3 **Objetivos de la Investigación.**

Objetivo general:

Desarrollo de un sistema de procesamiento y análisis de patrones en imágenes basado en el algoritmo SIFT (Scale Invariant Feature Transform) e implementarlo en el sistema operativo Android.

Objetivos específicos:

- Adquirir una serie de imágenes de los edificios de la UAQ, Campus Juriquilla en diferentes periodos de tiempo y variaciones ambientales.
- Crear una base de datos basada en las imágenes capturadas.
- Desarrollar una mejora al algoritmo SIFT.
- Implementar la mejora del algoritmo SIFT en el sistema operativo Android.
- Procesar las imágenes obtenidas en la implementación realizada en el sistema operativo Android.
- Analizar los resultados obtenidos con la implementación realizada y el algoritmo original.

1.4 Alcances y Limitaciones

Resultados generales que se pretenden con el proyecto de investigación.

- Implementación y programación de una mejora del algoritmo SIFT en el sistema operativo Android para el análisis y procesamiento de imágenes.
- Encontrar una posible solución para los problemas que se pueden presentar en cuestión del funcionamiento óptico del sistema debido a las variaciones del entorno.
- Desarrollar una aplicación que permita apreciar los resultados obtenidos en esta investigación.
- Publicar los resultados obtenidos en un Articulo de un congreso Internacional o Journal.

Participación en el proyecto de investigación.

La participación en este proyecto se enfocara primeramente a la adquisición de las imágenes que conformaran nuestra base de datos en la cual se realizaran las pruebas, se procederá a estudiar y programar el algoritmo SIFT y se realizaran las pruebas pertinentes, se analizaran los resultados obtenidos para saber cómo

se comporta y afronta los diversos factores anteriormente mencionados, como son variaciones de iluminación, enfoques, objetos irrelevantes, entre otros.

Se procederá a realizar una mejora en el algoritmo mediante un preprocesamiento en las imágenes para mejorar su desempeño en la extracción de características y su porcentaje de resultados correctos positivos, y se procederá a implementar en el sistema operativo Android.

Se realizara la documentación e investigación necesaria en cada uno de los procesos realizados.

1.6 Estructura de la Tesis

El trabajo de investigación constara de 6 capítulos y un anexo, en el 1° capítulo se presenta la definición del proyecto de investigación, posteriormente se presenta la justificación del trabajo realizado, así como los objetivos de la investigación, tanto generales como particulares, los alcances y limitaciones que se tienen en cuenta al realizar el trabajo y por ultimo una introducción a la forma en que estará organizado el trabajo.

En el capítulo 2, se describe el concepto de procesamiento de imágenes, así mismo se muestra el concepto de los algoritmos de visión por computadora, sus bases e inicios, asi mismo se mencionan algunos de los ejemplos más comunes utilizados de manera comercial e industrial, y se aborda la teoria y concepto del algoritmo SIFT.

En el capitulo 3, se describe la metodologia propuesta en esta investigacion, las mejoras que se proponen en base al algoritmo SIFT original tambine se incluyen las imágenes con las que se van a realizar los distintos preprocesamientos propuestos.

En el capitulo 4, se presentan los resultados obtenidos al realizar las diferentes pruebas, con el metodo propuesto y el algoritmo original, tambien se

incluye una tabla comparativa donde se muestran las diferencias en diferentes condiciones de iluminacion.

En el capitulo 5, se presentan las conclusiones finales para la investigacion realizada en base a las diferentes pruebas realizadas, y se realizan algunas recomendaciones para un trabajo a futuro para darle continuidad a la presente.

En el capitulo 6, se encuentra las referencias que fueron consultadas en el proceso de investigacion, desarrollo y conclusiones del proyecto de investigacion.

Finalmente en el capitulo 7, se incluyen los articulos publicados producto de la investigacion realizada.

2. Antecedentes, Trabajo Relacionado y Fundamentación Teórica.

2.1 Procesamiento de Imágenes

Uno de los principales procesos o etapas de las aplicaciones de visión por computadora es el procesamiento de imágenes, es decir, convertir o adaptar la imagen con que se desea trabajar en una más apta para el análisis, algunos ejemplos de esto es, una corrección de su exposición o balance de color, la reducción de ruido que pudiera dificultar o entregar resultados negativos en la imagen, incrementar la nitidez de la imagen, el rotamiento, el enderezamiento de la imagen mediante rotación, entre otros, todos estos procesos son esenciales en la mayoría de las aplicaciones de visión por computadora, incluso en fotografía computacional y reconocimiento, ya que en estas imágenes se requiere cierto cuidado en la fase del procesamiento de imagen para tener resultados aceptables y no obtener resultados positivos falsos.

2.2 Algoritmos de visión por computadora.

El objetivo de la visión por computadora es describir el mundo que nos rodea en una o varias imágenes y reconstruir las propiedades de estas, como lo es su forma, iluminación, color, entre otras, esto para el ojo humano es tan sencillo, a diferencia de los algoritmos de visión por computadora los cuales son muy propensos a errores (Richard, 2010).

Aun así hoy en día se han desarrollado una variedad de aplicaciones que se usan día a día, ya sea en un smartphone o en el mundo laboral, algunas aplicaciones comunes de los algoritmos de visión por computadora son: la extracción de estructuras a partir de una imagen con movimiento, estos algoritmos también puedes dar seguimiento a una persona mientras camina frente a un fondo poco ordenado (Han, 2013), estos algoritmos también pueden hacer detección de rostro, ya sea con detección de color en la ropa o detección de cabello, también pueden localizar y reconocer a los individuos en las imágenes. Algunos investigadores se han dedicado a desarrollar técnicas para recuperar la figura en tres dimensiones de un objeto, los que nos lleva a tener técnicas precisas para el

computo espacial 3d desde una serie de fotografías sobrepuestas. Podemos dar seguimiento a una persona moviéndose a través de un entorno complejo, incluso se puede lograr encontrar el nombre de las personas en una fotografía usando una combinación de su rostro, ropa y cabellos para el reconocimiento, pero aún así, todavía existen muchas limitantes y factores que pueden llevar al fallo de las aplicaciones de visión, por estas razones es complicado en ocasiones las aplicaciones de visión por computadora, ya que intentamos encontrar algo desconocido dando información insuficiente intentando encontrar una solución especifica.

Reconocimiento Óptico de Caracteres (OCR): El reconocimiento de caracteres escritos a mano o impresos en cartas, señales de tránsito, revistas, placas de automóvil, son algunas aplicaciones comunes en esta rama de la visión por computadora.

Inspección de Maquinaria: Inspecciones automatizadas en líneas de producción utilizando estéreo visión con iluminación especial para mediar las tolerancias en partes de aviones o vehículos, o la detección de defectos en los componentes mediante visión de rayos X.

Modelado 3D de edificios (Fotogrametría): construcción automática de modelos 3D a partir de fotografías aéreas.

Imágenes Medicas: registro de imágenes pre operación e intra operación, o realizar estudios a largo plazo de la morfología cerebral conforme se va envejeciendo.

Seguridad Automotriz: Detección de obstáculos inesperados o peatones en las calles, bajo condiciones donde la visión activa no funciona bien como lo es un radar

Emparejamiento de movimiento: Fusionando imágenes generadas por computadora con video en tiempo real, se puede lograr un seguimiento de características en un video para estimar el entorno 3D y la forma del entorno, esta

técnica es muy utilizada en películas en las cuales se insertan elementos inexistentes en la vida real al video.

Captura de movimiento: usando marcadores retroreflectivos vistos desde varias cámaras o algún otro sistema de visión se captura el movimiento del objeto o la persona para realizar una animación por computadora.

Vigilancia: Monitoreo de intrusos, análisis de tráfico vehicular, monitoreo de piscinas, entre otros.

Reconocimiento de huellas digitales y biométricas: para la automatización de la autentificación así como para aplicaciones forenses.

Existen múltiples aplicaciones mas de las ante4s mencionadas en la industria, también existen muchas en un ámbito más comercial o utilizadas por una usuario común, ya sea en un smartphone o en una computadora, algunas de las más comunes son:

Stitching: Creación de una foto panorámica a partir del superposicionamiento de varias fotografías.

Exposure Bracketing: Fusionando múltiples imágenes con diferente exposición, es decir con condiciones de iluminación poco favorables, es decir, demasiada iluminación o sombras, en una sola imagen con la exposición perfecta.

Morphing: cambiando una imagen gradualmente en otra utilizando transiciones morfológicas.

Modelado 3D: Convertir una o más imágenes en un modelo 3D de un objeto o una persona.

Emparejamiento de video y estabilización: Insertar imágenes en 2 dimensiones o modelos 3D en videos mediante el seguimiento de puntos de referencia, y el uso de movimiento estimado para remover las vibraciones.

Tutoriales/Guías basadas en fotografías: navegación por una amplia colección de fotografías desplazándose entre diferentes fotos de 3D.

Detección de Rostros: Para mejorar el enfoque de una cámara así como para realizar una mejor búsqueda de imágenes.

Autentificación Visual: autentificación automática mediante el rostro de un grupo de personas o familia en una computadora o dispositivo con solo estar frente a él.

Siempre se tienen que tomar en cuenta todos los factores posibles que puedan afectar la imagen, como fue creada la imagen, como interaccionan las luces con la escena y los efectos atmosféricos y como es el funcionamiento de los sensores, incluyendo las fuentes de ruido y incertinidad.

2.3 Detección de Características.

La detección de características es el primer paso para llegar a un emparejamiento de imágenes, lo que representa la base del modelado 3D. Existen muchos trabajos que apuntan a la detección de características y a su vez a el emparejamiento de imágenes, por lo que se considera importante este proceso. Las primeras implementaciones desarrolladas muestran que solo trabajaban bien con ciertas condiciones limitadas en las imágenes y el reto real de los autores fue el lograr un verdadera invariancia en la detección de características bajo cualquier condición (Iluminación, rotación, falta de definición, escala, oclusión), la consistencia de los resultados eran controlados mayormente por el tipo de imágenes usadas.

2.4 Detectores de Esquinas, Bordes y Textura.

La detección robusta de características, el emparejamiento de imágenes y los modelos 3D son conceptos que existían en el campo de la visión por computadora, pero no fue hasta inicios de la década antepasada que se empezaron a atacar estos problemas por numerosos investigadores de este

campo. Como es bien conocido, lograr un reconocimiento invariable de un objeto es uno de los principales retos en la visión pro computadora, hoy en día se ha logrado un progreso significante en el uso y la implementación de algoritmos mediante la detección de características invariables en imágenes más complejas (Lowe, 1999).

Al realizar un reconocimiento en un imagen digital se está limitado a la identificación de esquinas y bordes, esta practica aunque es efectiva, todavía tiene muchas limitaciones. El uso solo del reconocimiento de esquinas no es suficiente para la elaboración de modelos 3D o reconstrucción de objetos, estos incluyeron posteriormente otra clase de algoritmos enfocados en el emparejamiento de textura para tener un mejor desempeño.

2.5 **Algoritmos basados en el emparejamiento de características: Detectores de Esquinas y Bordes.**

Uno de los primeros trabajos sobre la detección de características fue el de Harris y Stephen (Harris, 1988), en el cual se introdujo el Detector de Esquinas Harris (Harris, 1988), en el cual se presentaba un nuevo método para la detección y extracción de características/puntos o esquinas. El descriptor presentado funcionaba bien si las imágenes cumplían con los requerimientos básicos, aunque debido a que el descriptor solo detectaba esquinas, los resultados carecían de conectividad entre las características o puntos encontrados, lo que presentaba una gran limitación para la obtención de un descriptor de nivel mayor, es decir, para superficies y objetos (Harris, 1988), por lo que los puntos detectados con este método no tenían la invariancia suficiente para hacer una reconstrucción 3D, pero aun así, este método de detección de esquinas ha sido usado ampliamente para diferentes aplicaciones especificas en el campo de la visión por computadora.

Imagen 2-1: Detección de bordes con el detector Harris

En 1988, Harris le dio otro enfoque a su trabajo ya realizado en el ámbito de la detección de características, conociendo las limitaciones de su trabajo previo, determino la necesidad de consistencia en las esquinas detectadas, un factor importante en la interpretación de imágenes 3D (Harris, 1988), por lograr esto combino las esquinas aisladas detectadas con el Detector de Harris con un borde de conexión correspondiente, de esta manera, las esquinas detectadas al azar por el Detector de Harris se asignaban a una geometría y espacio especifico lo que hacía más robusto el emparejamiento. Posteriormente a finales de 1990 surgieron varios detectores de bordes y equinas que estuvieron disponibles y publicados para el público en general, los más sobresalientes debido a su desempeño fueron el Detector de Esquinas SUSAN (Smith, 1997) y el método WANG(Wang, 1994).

2.5.1 **Detector de Esquinas FAST: Features from Accelerated Segment Test.**

Trajkovic y Hedley (Trajkovic y Hedley, 1998) publicaron en 1998 un nuevo algoritmo de detección de esquinas llamado FAST, en este trabajo se le dio una mayor importancia a las esquinas, ya que estas son las características más intuitivas que muestran un cambio de intensidad dimensional, por lo que son bien distinguidas entre los puntos vecinos (Trajkovic and Hedley, 1998). También mencionan estos autores que para que sea posible el emparejamiento de los puntos encontrados por los detectores de esquinas, se tiene que cumplir con ciertas condiciones:

- Consistencia en las características detectadas, es decir, deben ser invariantes al ruido y ser constantes si se obtienen múltiples imágenes de la misma escena.
- Precisión, las esquinas detectadas deben estar lo más cercanas a la posición correcta.
- Velocidad, tiene que ser lo suficientemente rápido el proceso de la detección.

Una de las principales ventajas del método propuesto por Trajkovic y Hedley (1998), fue el incrementar la velocidad de cálculo en la detección de las esquinas usando una función de respuesta de equinas (CRF) el cual asignaba un valor a la fuerza de la esquina basada en la intensidad local de los puntos vecinos en la imagen.

Este algoritmo demostró ser uno de los mejores comparado con los algoritmos anteriores en ese tiempo, Harris, Harris Modificado, SUSAN y Wang, logrando una mejor precisión en los resultados, aunque mostro un poco menos consistencia en los resultados que el método de Harris, FAST probo ser significativamente más rápido que cualquier otro algoritmo, (Trajkovic y Hedley, 1998).

Imagen 2-2: Detección de Esquinas con el algoritmo FAST.

En 2010 Fraser (Fraser, 2010) en su trabajo determino que una de las debilidades mas constates en los algoritmos de detección de esquinas, esta es la falta de efectividad cuando se quiere detectar una esquina en una imagen con una gran de agrupamiento de objetos, esto debido a que los detectores de esquinas están basados en el análisis de un pixel y sus pixeles vecinos, sin ningún proceso adicional de filtrado lo que lleva en ocasiones a detecciones erróneas, para esto se puede aplicar una aproximación lineal entre pixeles para reducir la sensibilidad del algoritmo a las esquinas falsas en regiones con textura de una imagen.

En el 2000, Fraser, (Fraser, 2010) combinaron el algoritmo FAST y otros algoritmos de detección de bordes y esquinas con otros procesos computacionales como un filtro Wallis, haciendo mucho más efectivo el algoritmo para lograr un emparejamiento y reconstrucción de imágenes 3D.

2.6 Algoritmos basados en Textura.

Otro reto que se presenta en los algoritmos de visión pro computadora es lograr un emparejamiento confiable en imágenes con mucha textura o con fondos desordenados, o con gran cantidad de objetos similares o con mucha textura.

Estos algoritmos han sido ampliamente usados como detectores de puntos de interés, debido a que las esquinas y los bordes corresponden a la localización

de imágenes con gran contenido de información, es decir, se pueden emparejar imágenes entre sí de manera confiable (Trajkovic,1998).

Sin embargo, los algoritmos basados en el reconocimiento de textura solo tienen un desempeño preciso cuando el objeto a emparejar tiene esquinas o bordes distintivos, es decir, solamente son adecuados cuando el emparejamiento es entre superficies planas y objetos dentro de la imagen dada. tampoco se obtienen los resultados deseados cuando las imágenes son sometidas a variaciones en escala, iluminación, rotación o transformaciones afines.

Debido a las limitaciones presentadas, se desarrollaron los algoritmos basados en Textura debido a su capacidad de emparejar características entre diferentes imágenes sin importar la presencia de un fondo texturizado o la falta de bordes planos o bien definidos. Uno de los primeros intentos en esta nueva propuesta de algoritmos fue llevada a cabo por Lowe (Lowe, 1999), proponiendo uno de los métodos mayormente reconocidos y usados en el ámbito de los algoritmos de emparejamiento basado en textura. Para que las características sean debidamente detectadas estas son filtradas por etapas con lo cual se pueden identificar puntos estables en el espacio escala.

2.6.1 **La propuesta de David Lowe.**

El trabajo presentado por Lowe (Lowe, 1999) demostró que era posible detectar características invariantes a escala, traslación, rotación y parcialmente invariables a cambios en la iluminación. Al inicio de esto, son creadas las imágenes de referencia que darán paso a las deformaciones locales representadas por los gradientes de las imágenes en múltiples planos de orientación y múltiples escalas, asegurando la detección de puntos incluso en objetos con fondos heterogéneo. Estas imágenes de referencia son creadas en el proceso de filtrado, y son usadas como entrada al indexar el método del vecino más cercano, el cual identifica los objetos candidatos a ser emparejados.

Para hacer una verificación final de emparejamiento, esto se lleva a cabo encontrando un residuo bajo de una solución de mínimos cuadrados para los parámetros desconocidos de los modelos, de esta manera se pueden obtener resultados en imágenes con oclusión parcial.

2.6.2 La propuesta de Mikolajczyk y Schimd.

Mikolajczyk and Schmid presentaron en el 2004 un nuevo enfoque para detectar puntos de interés invariantes a escala y afines a transformaciones (Mikolajczyk, 2004). Su trabajo combina el detector de Harris (Harris, 1988) con Selección de escala basada en Laplacianos de Lowe (1999), afrontando el problema de la invariancia a las transformaciones afines y presentaron un nuevo detector de características que seleccionaba puntos desde una representación multi escala (Mikolajczyk, 2004). Esta metodología logro la detección de características invariantes a transformaciones afines debido a que era adecuado para trabajar con imágenes con escalas no uniformes, a diferencia de los detectores de características anteriores. Posteriormente Schimd propuso en colaboración con Dorkó (Dorkó y Schmid, 2003), para construir y seleccionar partes de objetos invariantes a escala, es decir, en lugar de seleccionar características en una imagen completa, el descriptor solo las extraía de objetos específicos en un set de imágenes.

2.7 Algoritmo SIFT (Scale Invariant Feature Transform).

Posteriormente a la publicación de Lowe (Lowe, 1999) sobre la detección de características en imágenes texturizadas, presento una versión mejorada de los resultados de su trabajo previo, así mismo, presento el algoritmo titulado Scale Invariant Feature Transform (SIFT) (Lowe, 2004). Este algoritmos es un método para detectar en imágenes características distintivamente invariantes que posteriormente pueden ser utilizadas para hacer un emparejamiento entre diferentes vistas del objeto o escena, esto se puede lograr debido a que el algoritmo se enfoca en el uso de un filtro en forma de cascada para detectar las

características lo que transforma los datos de la imagen en coordenadas invariantes a la escala relativas a las características locales.

El algoritmos SIFT consta de cuatro etapas principales, las cuales son mencionadas a continuación:

- Construcción de Pirámide Scale-Space y Detección de Extremos.
- Localización de Puntos Clave
- Asignación de Orientación
- Descriptor de Puntos Clave

Cada uno de estos pasos es ejecutado en orden descendente, por lo cual se hacía referencia a la forma de cascada, en cada una de las etapas se realiza proceso de filtrado solo de esta manera los puntos clave o de interés son lo suficientemente robustos para pasar al siguiente proceso.

2.7.1 **Construcción de Pirámide Scale-Space y Detección de Extremos.**

Se representa la imagen en diferentes escalas y tamaños. Se lleva a cabo de manera eficiente mediante el uso de la función de diferencia Gaussiana, para identificar los posibles puntos de interés que son invariables a escala y orientación.

Para la detección de extremos locales se debe trabajar con la imagen original filtrada. El único filtro apropiado para estos efectos son los filtros Gaussianos pasa bajas. Se utiliza este tipo de filtros debido a que la función Gaussiana es invariante a escala en el espacio (Imagen 2-3), para la detección de puntos de interés. Además, elimina el ruido de la imagen.

Imagen 2-3: Imagen en diferentes escalas

La imagen con la que se trabaja es la convolución entre la imagen original y el filtro Gaussiano. Si llamamos la imagen original I(x, y) y al filtro G (x, y, σ), entonces la imagen con la que se trabaja es L(x, y, σ) que resulta de la convolución entre ambas, es decir:

$$L(x,y,\sigma) = G(x,y,\sigma) * I(x.y) \tag{1}$$

$$G(x,y,\sigma) = \frac{1}{\sqrt{2\pi\sigma}} * \left(-\frac{x^2+y^2}{2\sigma^2}\right) \tag{2}$$

Adicionalmente, se utiliza una distinta desviación estándar para el filtrado de la diferencia de dos filtros Gaussianos, cuya diferencia es un factor k.

$$D(x,y,\sigma) = G(x,y,k\sigma) - G(x,y,\sigma) \tag{3}$$

$$L(x,y,\sigma) = D(x,y,\sigma) = D((x,y,\sigma) * I(x,y) \tag{4}$$

Otra manera de trabajar estas imágenes es utilizando el Laplaciano de funciones Gaussianas, sin embargo, la utilización de esto es lenta y la diferencia de Gaussianas es una aproximación bastante eficiente. El Laplaciano tiene la formula con G como se definió anteriormente (Fernandez, 2009).

$$M = \sigma^2 * \Delta G = \sigma^2 \left(\frac{\partial^2 G}{\partial x^2} + \frac{\partial^2 G}{\partial y^2}\right) \tag{5}$$

En una imagen real, las frecuencias que contienen la información característica son las medias. El ruido digital está en las frecuencias altas, que está en cada imagen digital. Por otro lado, las frecuencias bajas, contienen variaciones suaves por lo que no son relevantes para la obtención de los puntos de interés. Como se puede ver fácilmente, la aproximación del las dos fórmulas sigue de los cálculos siguientes usando la definición de la derivada:

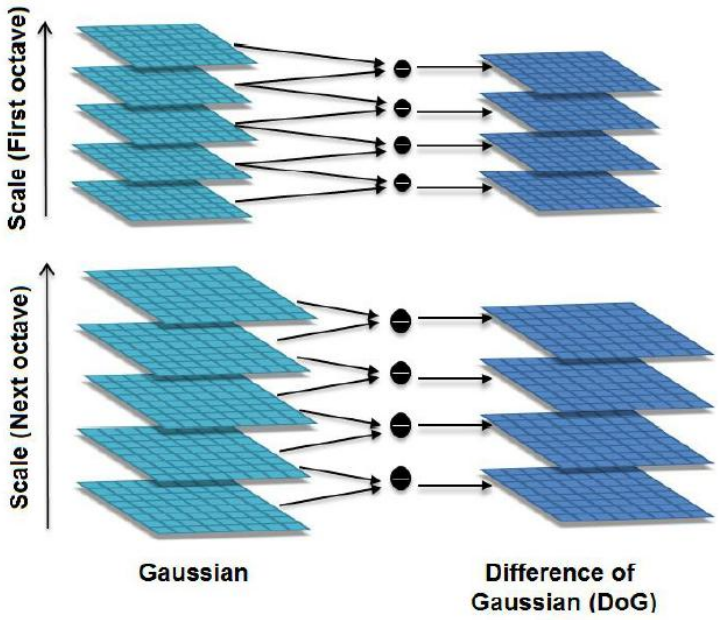
$$\sigma * \Delta G = \frac{\partial G}{\partial \sigma} \approx \frac{G(x,y,k\sigma) - G(x,y,\sigma)}{\sigma(k-1)} \tag{6}$$

$$\rightarrow G(x,y,k\sigma) - G(x,y,\sigma) \approx$$

$$\approx (K-1) * \sigma^2 * \Delta G \tag{7}$$

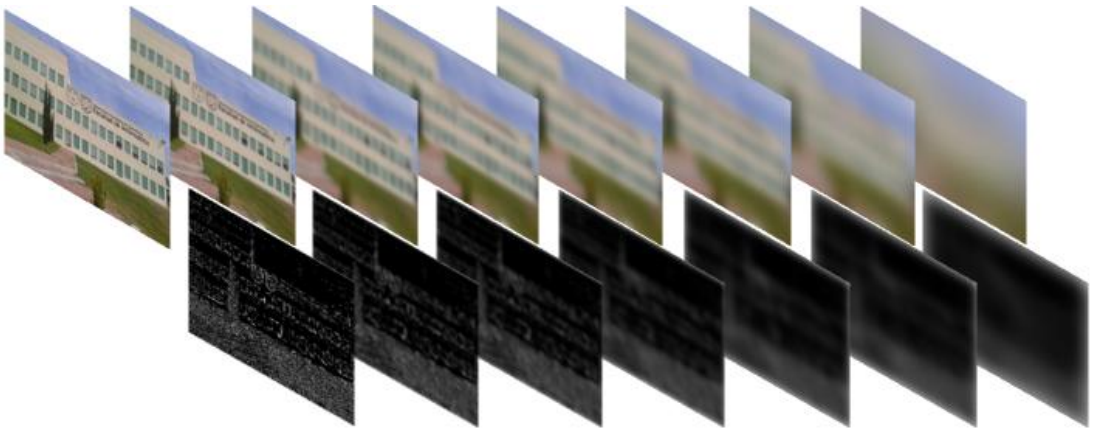
La convolución con la DoG (Diferencia Gaussiana) se hace para toda la imagen y con escalas diferentes, para detectar estructuras en todos los lugares y con todos los tamaños. Para una implementación rápida, el algoritmo trabaja cada octava en forma individual, como vemos en la Imagen 2-5.

Imagen 2-4: Por cada octava en las diferentes escalas se aplica una diferencia Gaussiana (DoG).



Como menciona Lowe (Lowe, 1999, 2004) en sus investigaciones, para cada octava un número de 3 escalas y un sigma de 1.6 eran los valores óptimos. Por eso, primero se hace la convolución con 3 escalas de las Gaussianas, y luego para obtener la DoG se hace con la resta de imágenes vecinas como se puede observar en la Imagen 2-6 donde se ve el resultado que se obtiene tratando imágenes reales.

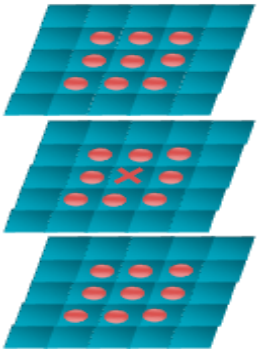
Imagen 2-5: Diferencia de Gaussiana en imágenes reales.



Para la octava siguiente, solamente hay un nuevo muestreo con un factor 2 y la repetición de la resta. Con este método se crean muchas imágenes filtradas con valores extremos donde el tamaño y el lugar de la DoG es similar a la estructura dentro de la imagen.

Para buscar los extremos en las imágenes convolucionadas, cada píxel es comparado con todos sus píxeles vecinos, ambos en el dominio del espacio y en del dominio de la escala (Imagen 2-7). Solo si todos tienen un valor distinto, este lugar va a pasar el examen.

Imagen 2-6: Cada uno de los pixeles de la imagen es comparado con sus pixeles vecinos en la pirámide de diferencia de Gaussianas.



2.7.2 Localización de Puntos Clave.

Entre los puntos que sobrevivieron el examen de la búsqueda de extremos hay muchos que caracterizan puntos con poco contraste. Estos no son estables, si la iluminación cambia producen ruido. Para quitarlos se examina primero si el máximo o mínimo está en un lugar entre esos píxeles, para estimar la función D con una serie de Taylor de grado 2.

$$D(x) = D + \frac{\partial D^T}{\partial x} x + \frac{1}{2} x^T \frac{\partial^2 D}{\partial x^2} x \tag{8}$$

Después de la derivación de esta aproximación e igualando a cero queda:

$$\hat{x} = -\frac{\partial^2 D^{-1}}{\partial x^2} * \frac{\partial D}{\partial x} \tag{9}$$

$$\rightarrow D(\hat{x}) = D + \frac{1}{2} * \frac{\partial D^T}{\partial x} \hat{x} \tag{10}$$

Si el valor de D(x^) es menor a 0.03, el punto es eliminado, suponiendo que D tiene valores de 0 a 1. Además de quitar aquellos puntos con poco contraste, hay que encontrar y descartar candidatos que vienen de una línea recta y no de una esquina. Si hay una línea recta, la curvatura de D va a ser grande en una dirección pero pequeña en la que es perpendicular. Este factor corresponde a un valor propio α grande y un β más pequeño de la matriz Hessiana.

$$\mathrm{Traza(H)} = \frac{\partial^2 \mathrm{D}}{\partial \mathrm{x}^2} + \frac{\partial^2 \mathrm{D}}{\partial \mathrm{y}^2} = \alpha + \beta \tag{11}$$

$$\mathrm{et(H)} = \frac{\partial^2 \mathrm{D}}{\partial \mathrm{x}^2} * \frac{\partial^2 \mathrm{D}}{\partial \mathrm{y}^2} = \alpha * \beta, \alpha = \mathrm{r}\beta \tag{12}$$

Para un r predefinido hay que examinar si se cumple la inecuación en la que el trabajo de Lowe (Lowe , 1999, 2004) propone un umbral de r = 10:

$$\frac{\mathrm{Traza(H)}^2}{\mathrm{Det(H)}} < \frac{(\mathrm{r}+1)^2}{\mathrm{r}} \tag{13}$$

2.7.3 **Asignación de Orientación.**

En este paso se asigna una dirección a cada punto de interés el cual depende de las muestras de los puntos que se poseen en su entorno.

Para tener un buen descriptor, la localización y la función local de aproximación que tenemos hasta ahora no son suficientes. Hay también que examinar el valor del gradiente y su orientación. El valor corresponde a la escala de la Gaussiana y mediante ese tratamiento la descripción del punto de interés es invariante con respecto a la escala. Además con el conocimiento de la orientación, la caracterización es independiente con respecto a la dirección. En un espacio vectorial de Euclides, se calcula el valor de la Longitud del Gradiente m(x, y) y su Orientación Θ(x, y) de una función discreta como:

$$\mathrm{m(x,y)} = \sqrt{(\Delta \mathrm{L_x})^2 + (\Delta \mathrm{L_y})^2} \tag{14}$$

$$= \sqrt{(\mathrm{L(x+1y) - L(x-1y)})^2 + (\mathrm{L(xy+1) - L(xy-1)})^2} \tag{15}$$

$$\theta(\mathrm{x,y}) = \tan^{-1}\left(\frac{\Delta \mathrm{L_y}}{\Delta \mathrm{L_x}}\right) \tag{16}$$

$$= \tan^{-1}\left(\frac{\mathrm{L(x,y+1) - L(x,y-1}}{\mathrm{L(x+1,y) - L(x-1,y)}}\right) \tag{17}$$

Para tener un conocimiento más general de la orientación en un punto de interés, el algoritmo SIFT examina también los valores m y Θ de los píxeles que

están en su entorno. Gracias al uso de una ventana Gaussiana, los valores de píxeles más lejanos tienen un impacto más pequeño que píxeles cercanos. Con un histograma de orientación con ventanas de 10 grados el algoritmo trata de buscar la dirección verdadera usando una interpolación de los 3 valores más grandes del histograma. El histograma de orientación es formado por la orientación del gradiente de los puntos muestreados en la región del punto clave. Cada pico en el histograma corresponde a la dirección dominante del gradiente local (un pico con el 80% es usado para crear el punto clave con su orientación). Sólo alrededor del 15% de los puntos se les asignan múltiples orientaciones, en la Figura 10 se aprecia una representación de los puntos clave y su orientación.

Imagen 2-7: Representación de puntos clave con sus respectivas direcciones, se puede apreciar que no importa la rotación de la imagen, los puntos siguen siendo identificados correctamente (Yu, 2010).

2.7.4 **Descriptor de Puntos Clave**

Para tener una descripción que sea lo adecuadamente invariante a cambios en un punto de vista en el espacio, un tratamiento relacionado a la función de las neuronas en la visión biológica es usado. Debido a que es necesaria la independencia de pequeñas translaciones del punto de interés, Edelman (Edelman,1997) propuso el algoritmo siguiente.

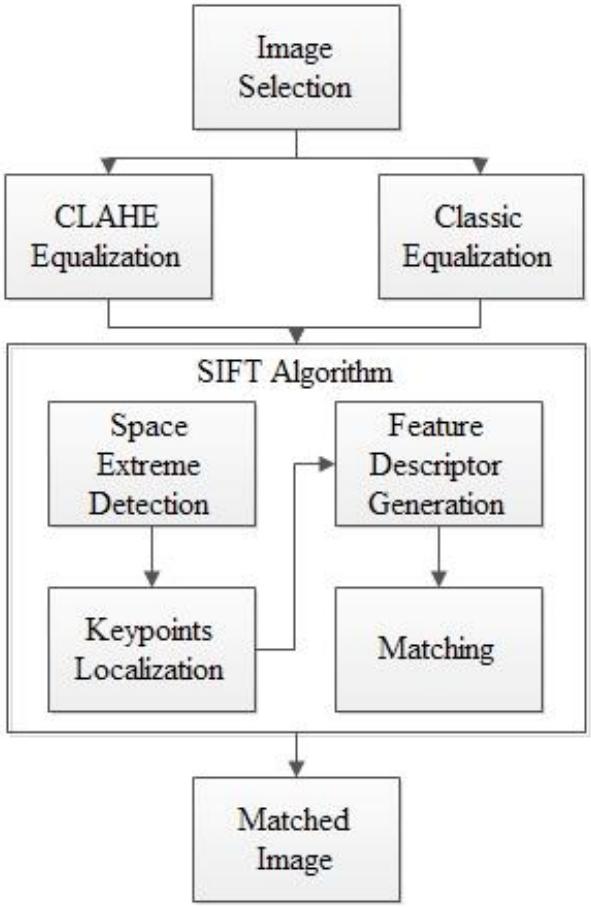
Imagen 2-8: Usando una ventana Gaussiana, los valores m y Θ se examinan en la vecindad del punto de interés, después sigue un tratamiento de histogramas para 8 orientaciones distintas.

Una ventana Gaussiana representada por el círculo en la Imagen 2-9, selecciona los valores de m y Θ prefiriendo los que están más cerca del centro. Después sigue una distribución en sectores más grandes y otra vez el uso de un histograma con 8 distintas orientaciones. La ventaja de esto es que los histogramas quedan iguales, aún cuando el centro de la ventana Gaussiana se mueve hasta 4 píxeles. Esto hace la descripción bastante robusta con respecto a las translaciones por cambios de puntos de vista. La Figura 8 muestra un ejemplo reducido a 2x2 histogramas. Según el algoritmo SIFT se tratará con 4x4 con 8 posibles direcciones correspondiente a un vector de 4x4x8 = 128 dimensiones para cada punto de interés.

3. METODOLOGÍA

Para la propuesta realizada en este trabajo de investigación se ha optado por agregar una etapa de preprocesamiento mas al algoritmo SIFT propuesto por Lowe (Lowe, 2004), para este preprocesamiento se utilizaron dos técnicas, la primera fue una ecualización de histograma y la segunda fue utilizando el algoritmo Contrast Limited Adaptive Histogram Equalization (CLAHE) (Yussof, 2013), el cual es una mejora a la ecualización de histogramas tradicional basada en un manejo de contrastes variables para una mejor respuesta de la imagen a realzar (Palma, 2014).

Imagen 3-1: Metodología Propuesta.



3.1 Imágenes Utilizadas en el Estudio

Para realizar el estudio se seleccionaron una serie de imágenes de los edificios de la Facultad de Informática Campus Juriquilla de la Universidad Autónoma de Querétaro.

Las muestras se tomaron en diferentes momentos del día y con diferentescondiciones de iluminación natural e iluminación artificial cuando se requería (Takacs, 2008).

Imagen 3.2: Parte representativa de la base de datos de imágenes obtenida de uno de los edificios de la Facultad de Informática de la Universidad Autónoma de Querétaro en las que se puede apreciar el cambio de iluminación a lo largo del día.

De las imágenes obtenidas en la base de datos se procedió a selecciona un par de imágenes que serian utilizadas como el patrón de referencia al cual se le aplicara el procesamiento de imágenes para posteriormente aplicar el algoritmo SIFT y ver el comportamiento que tiene la imagen preprocesada.

3.2 Software propuesto para la Investigación.

Para realizar el preprocesamiento y las diferentes implementaciones del algoritmo SIFT se decidió trabajar con librerías dedicadas al ámbito de visión por computadora, la primera de ellas es la librería de OpenCV para los lenguajes de programación C++ y Java, la segunda librería utilizada es el toolbox Balu en conjunto con la librería VLFeat, estas dos para el software de lenguaje de alto nivel Matlab, se seleccionaron esta librerías ya que cuentan con muchas funciones ya implementadas para el procesamiento y análisis de imágenes y por otra parte, son de código abierto, por lo cual las hace más flexibles y manipulables.

3.3 Obtención del Histograma de las Imágenes.

El histograma de una imagen es una representación grafica de los distintos valores de gris en una imagen dentro del rango $[0, L-1]$, es una funcion discreta $h(r_k) = n_k$, donde $r_k$es el $k - esimo$ nivel de gris y n_k es el numero de pixeles de la imagen que que tienen el nivel de gris r_k.

El histograma es la base de numerosas técnicas de procesamiento de la imagen en el dominio espacial (Tang, 2012), es decir, la manipulación del histograma puede utilizarse eficazmente para el realce de la imagen, por lo tanto, con ayuda del las funciones referentes al manejo de histogramas ya contenidas en las librerías antes mencionadas, se procedió a obtener los histogramas de las imágenes seleccionadas como patrón para analizar su distribución y ver las diferencias en la cantidad de tonalidades y la repetibilidad de cada uno de estos en la escala de grises.

El eje horizontal corresponde a los valores de gris r_k llendo desde el negro puro (valor 0) hasta el blanco puro (valor 255), el eje vertical corresponde a los valores de $h(r_k) = n_k$ que representa el numero de pixeles que contiene la imagen para cada tono.

Imagen 3-3: Imagen de día en escala de grises tomada como patrón (arriba), histograma obtenido de la imagen original sin ningún tipo de realce(abajo).

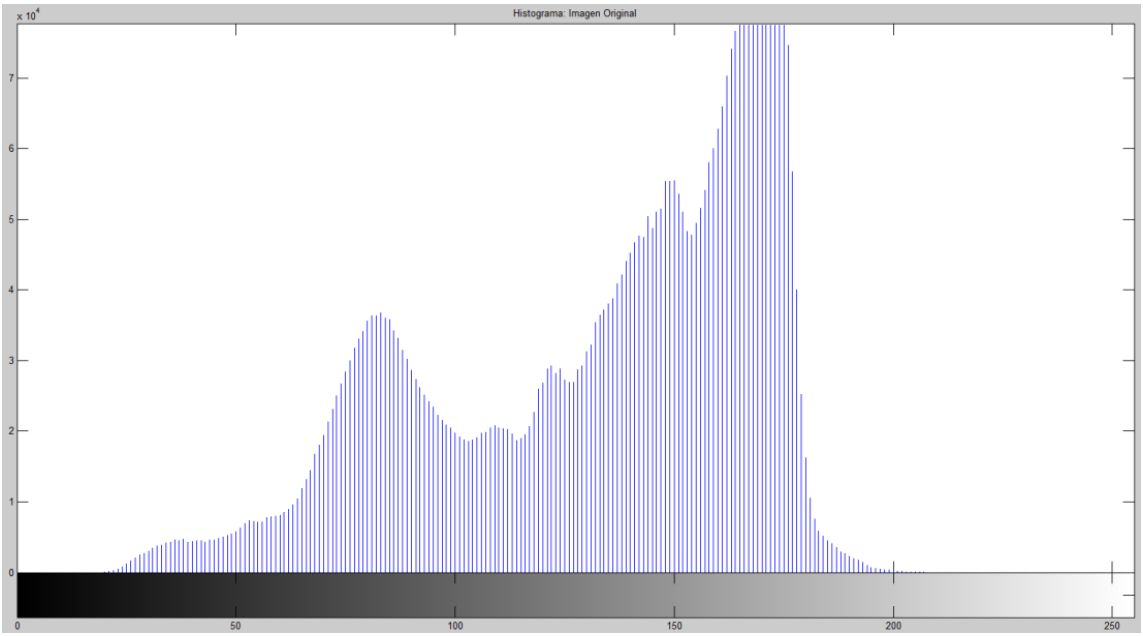
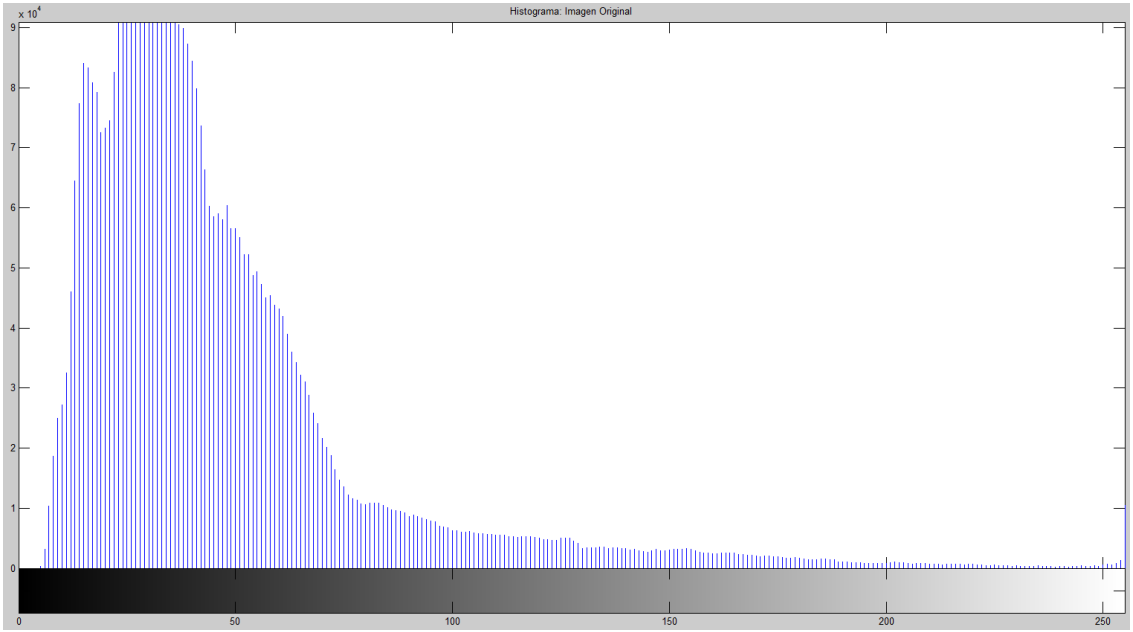


Imagen 3-4: Imagen de noche en escala de grises tomada como patrón (arriba), histograma obtenido de la imagen original sin ningún tipo de realce(abajo).



3.4 Ecualización de Histograma.

Posteriormente, a las mismas imágenes seleccionadas para el proceso anterior, se les aplico una ecualización tradicional analizar la nueva distribución con la que cuentan los histogramas realzados (Dalal, 2005).

Imagen 3-5: Imagen tomada en el día con un procesamiento de ecualización tradicional y su histograma obtenido de la imagen.

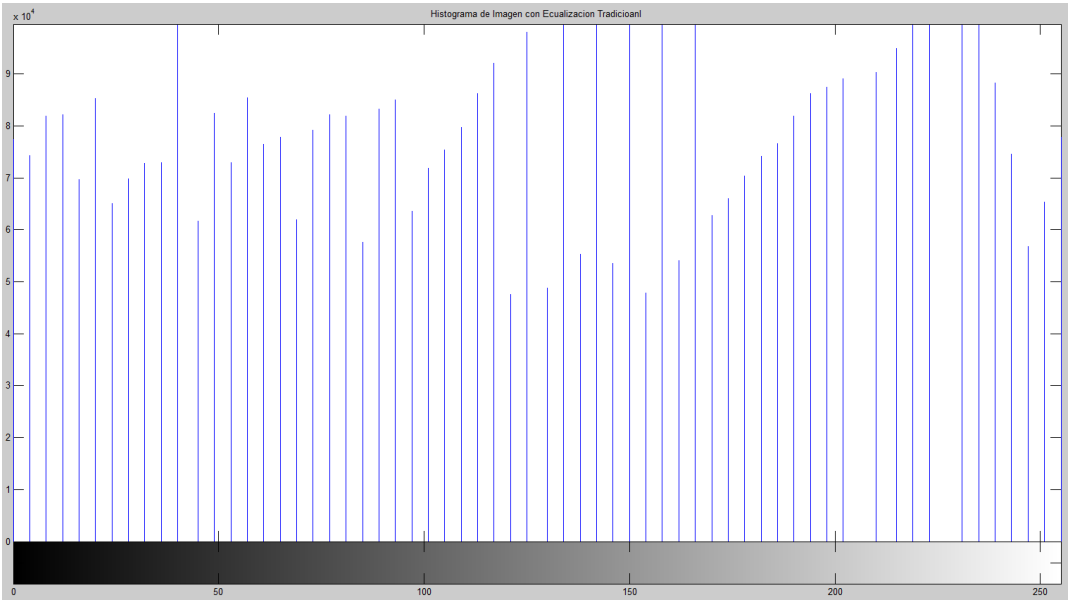
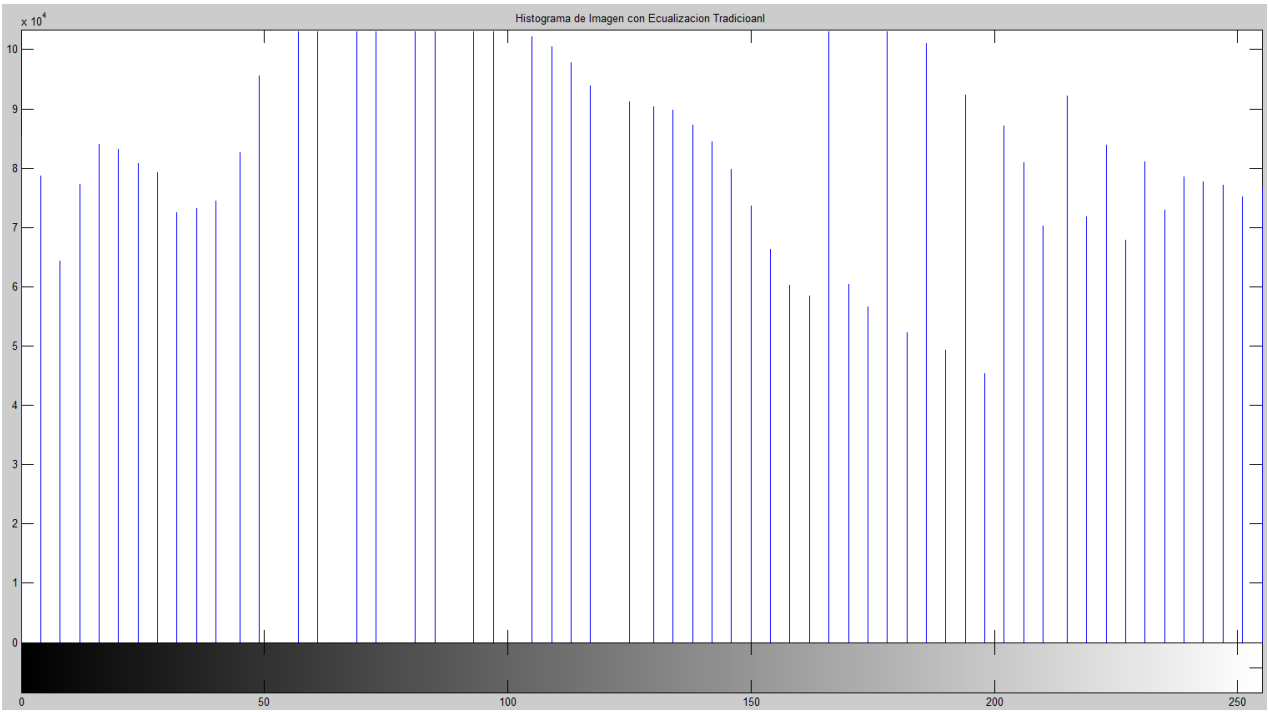


Imagen 3-6: Imagen tomada en la noche con un procesamiento de ecualización tradicional y su histograma obtenido de la imagen.



3.5 **Algoritmo CLAHE.**

Mientras que la ecualización global de histogramas puede ser útil en muchos casos, en algunas imágenes es preferible aplicar diferentes tipos de ecualización en diferentes regiones, es decir, en lugar de calcular una sola curva , se divide la imagen en bloques de $M\,xM$ pixeles, para realizar de manera separada la ecualización de histogramas en cada sub bloque. esto puede causar discontinuidades en los límites de los bloques. Una manera de eliminarla estas diferencias entre los bloques es usando una ventana móvil, es decir, recalcular el histograma para cada bloque $M\ x\ M$ centrado en cada pixel. Esto puede llegar a requerir más tiempo de computo, aunque se puede hacer la actualización solamente con los pixeles correspondientes a cada entrada y salida de los diferentes bloques.

Una manera más eficiente de hacer esta aproximación es calcular una función de ecualización basada en bloques no sobrepuestos, y después aplicar una interpolación de suavizado para la función de transferencia al movernos entre los bloques. A este proceso se le llama Ecualización de Histogramas adaptativos(AHE, por sus siglas en ingles), y la versión con limite de contraste es conocida como Ecualización de Histogramas Adaptativos con Limite de Contraste (CLAHE).

La función de ponderación para un pixel dado (i,j) pueden ser calculados como la función de su posición vertical y horizontal (s,t) dentro de un bloque, para mezclar las cuatro funciones de búsqueda $\{f_{00},\dots,f_{11}\}$ como una funcion de mezcla bilineal.

$$f_{s,t}(I) = (1-s)(1-t)f_{00}(I) + s(1-t)f_{10}(I) + (1-s)tf_{01}(I) + stf_{11}(I) \quad (18)$$

Nuevamente a la serie de imágenes que se seleccionaron para ser las imágenes de referencia o patrón se les aplico un realce de histogramas (Tu, 2013), ahora mediante el algoritmo CLAHE a cada una de las imágenes de los

diferentes momentos del día, seleccionando una distribución del tipo uniforme y obteniendo los siguientes histogramas:

Imagen 3-7: Imagen tomada en el día con un realce de histogramas mediante el algoritmo CLAHE (Arriba) y su histograma (Abajo).

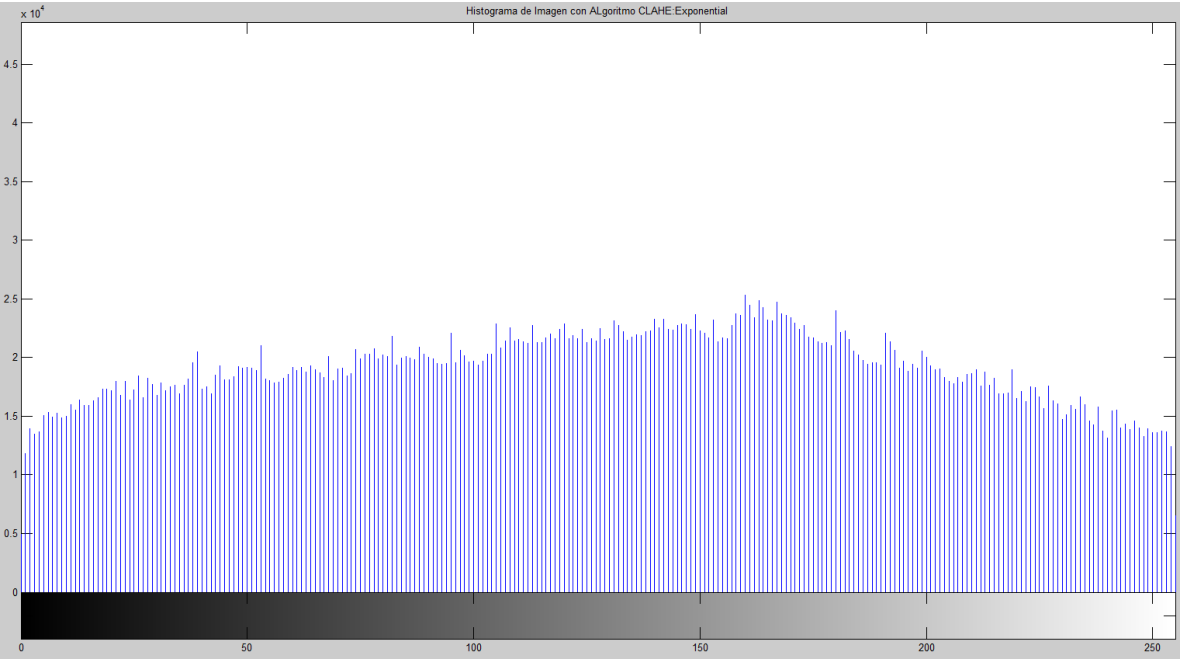
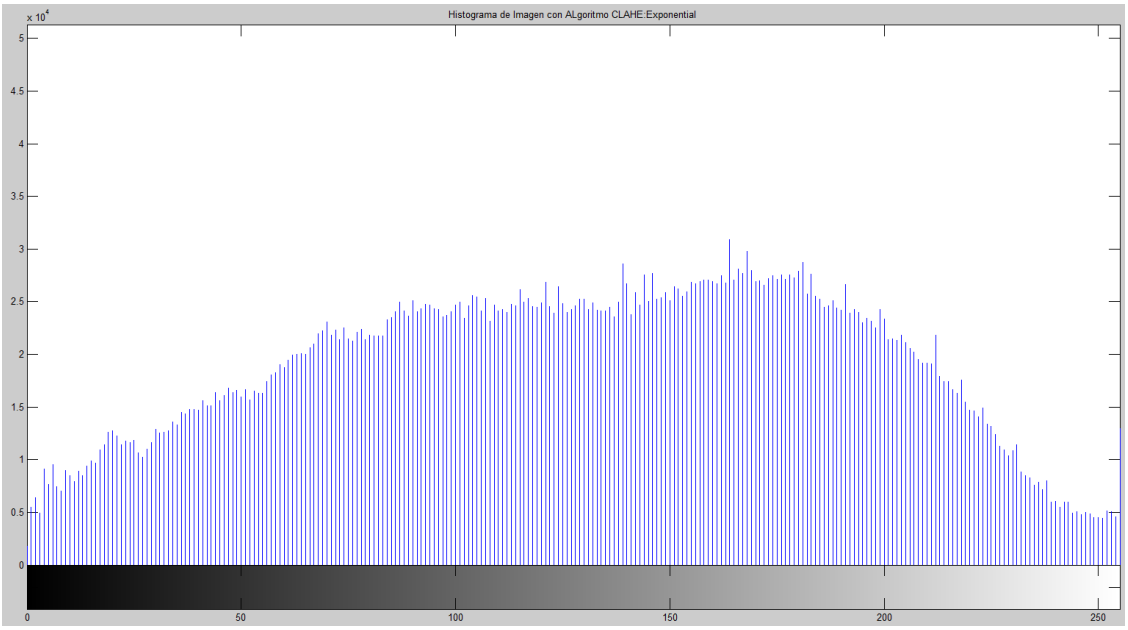


Imagen 3-8: Imagen tomada en la noche con un realce de histogramas mediante el algoritmo CLAHE (Arriba) y su histograma (Abajo).



3.6 Extracción de Características Mediante el Algoritmo SIFT.

Una vez que se tenían las imágenes con el debido preprocesamiento, estas se les aplico el algoritmo SIFT para hacer la extracción de características, es decir, los puntos importantes en cada una de las imágenes para posteriormente poder

realizar un emparejamiento con el resto de las imágenes de la base de datos y poder identificar de que edificio se trata.

Imagen 3-9: Representación de las características extraídas en una imagen de día (Arriba), una imagen por la tarde (Izquierda) y una imagen de noche (derecha) del mismo edificio.

Tabla 3-1: Cantidad de características extraídas en cada una de las imágenes patrón.

Tiempo de la Imagen Patrón	Características Extraídas		
	Imagen Original	Imagen Ecualizada	Imagen con Algoritmo CLAHE
Día	1207	2168	6343
Tarde	1940	2354	5937
Noche	136	1189	11489

3.7 Reconocimiento de las Imágenes.

Una vez extraídas las características se procede a hacer el emparejamiento entre las imágenes de la base de datos con las imágenes patrón que previamente se les aplico el procesamiento basado en realce de histogramas. Para ello también se hace uso del algoritmo SIFT extrayendo de la imagen de prueba sus características, se hace la comparación y emparejamiento entre las características de las dos imágenes para ver qué porcentaje de características iguales existen y en base la cantidad poder identificar si se trata del mismo edificio o se tratan de estructuras diferentes.

Imagen 3-10: Edificio reconocido por el algoritmo durante el día con la mejora propuesta del algoritmos SIFT.

4. Resultados y Analisis.

Se obtuvieron resultados satisfactorios al realizar las diferentes pruebas del algoritmos SIFT agregándole una etapa de preprocesamiento a las imágenes utilizadas como patrón. Se pudo apreciar notablemente el aumento en la cantidad de características extraídas en cada uno de los diferentes procesos de realce de histogramas, mostrando una mayor cantidad al aplicar el preprocesamiento basado en el algoritmo CLAHE.

Tabla 4-2: Cantidad de características emparejadas con el mejor y el peor de los casos de las imágenes de prueba a lo largo del día de la base de datos.

Imagenes Utilizadas		Imágenes Originales: Día		Imágenes Originales: Noche		Imágenes Preprocesadas: Día		Imagenes Preprocesadas: Noche	
		Patron 1	*Patron 2*	*Patron 1*	*Patron 2*	*Patron 1*	*Patron 2*	*Patron 1*	*Patron 2*
Noche	Mejor Caso	46	64	10	22	116	141	64	83
	Peor Caso	15	16	10	11	19	17	32	42
Tarde	Mejor Caso	57	46	10	18	39	25	20	50
	Peor Caso	22	15	8	10	15	19	14	15
Día	Mejor Caso	293	247	10	25	166	199	55	69
	Peor Caso	184	189	12	14	128	161	51	57

Ya que al usar dicho algoritmo se lograba obtener una cantidad notoria de características en imágenes carentes de iluminación, es decir, en imágenes de los edificios tomadas en la noche.

Una vez que se obtuvieron los resultados esperados, se procedió a importar el algoritmo ya realizado al lenguaje nativo del sistema operativo Android, Java siendo compatibles las diferentes librerías que se utilizaron en el proceso, la implementación se logro hacer sin muchos cambios o contratiempos en la programación.

Imagen 4-1: Implementación del algoritmo SIFT en Android y Reconocimiento del Edificio

5. Conclusiones y trabajo futuro.

A pesar del resultado un poco más favorable obtenido por la ecualización tradicional de histogramas aplicado a las imágenes utilizadas como patrón, se opto por utilizar el algoritmo CLAHE para tener un mejor resultado por sus propiedades adaptativas de contraste, que se adecuan de mejor manera para las condiciones de las fotografías tomadas en un ambiente no controlado al aire libre. Aunque en ocasiones se puede presentar una cantidad significativa de ruido debido a la ecualización, el algoritmo SIFT es capaz de filtrar y eliminar esa interferencia para evitar tener falsos positivos que puedan llegar a un reconocimiento erróneo.

6. REFERENCIAS.

Lowe, D. G. (1999). Object recognition from local scale-invariant features. In Computer vision, 1999. The proceedings of the seventh IEEE international conference on (Vol. 2, pp. 1150-1157). Ieee.

Tu, L., & Dong, C. (2013, December). Histogram equalization and image feature matching. School of Science, University of Science and Technology Liaoning, Qianshan, road no. 185 of An-Shan city, China. In Image and Signal Processing (CISP), 2013 6th International Congress on (Vol. 1, pp. 443-447). IEEE.

Edelman, S., Intrator, N., & Poggio, T. (1997). Complex cells and object recognition.

Dorkó, G., & Schmid, C. (2003, October). Selection of scale-invariant parts for object class recognition. In Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on (pp. 634-639). IEEE.

Palma O. R. D., Martínez Zerón, E., Aceves F. M. A., Tovar. S., Ramos Arreguín, J. M., & Pedraza Ortega, J. C. (2014) Analysis and comparison of images using the SIFT algorithm with a pre-processing stage based on histogram with different lighting conditions. In Engineering and its applications Vol. 1 on 10th International Engineering Congress. 2014. pp. 21-27.

Mikolajczyk, K., & Schmid, C. (2004). Scale & affine invariant interest point detectors. International journal of computer vision, 60(1), 63-86.

Dalal, N., & Triggs, B. (2005, June). Histograms of oriented gradients for human detection. In Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on (Vol. 1, pp. 886-893). IEEE.

Tang, H., & Tang, F. (2012, September). AH-SIFT: Augmented Histogram based SIFT descriptor. In Image Processing (ICIP), 2012 19th IEEE International Conference on (pp. 2357-2360). IEEE.

Mikolajczyk, K., & Schmid, C. (2005). A performance evaluation of local descriptors. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 27(10), 1615-1630.

Trajković, M., & Hedley, M. (1998). Fast corner detection. Image and vision computing, 16(2), 75-87.

Fraser, C. S., Jazayeri, I., & Cronk, S. (2010). Automated 3D object reconstruction via multi-image close-range photogrammetry. Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci, 38, 305-310.

Wang, H., & Brady, M. (1995). Real-time corner detection algorithm for motion estimation. Image and Vision Computing, 13(9), 695-703.

Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. International journal of computer vision, 60(2), 91-110.

Smith, S. M., & Brady, J. M. (1997). SUSAN—a new approach to low level image processing. International journal of computer vision, 23(1), 45-78.

Yussof, W. N., Hitam, M. S., Awalludin, E. A., & Bachok, Z. (2013). Performing Contrast Limited Adaptive Histogram Equalization Technique on Combined Color Models for Underwater Image Enhancement. International Journal of Interactive Digital Media, 1(1).

Alonso-Fernandez, F., Tome-Gonzalez, P., Ruiz-Albacete, V., & Ortega-Garcia, J. (2009, September). Iris recognition based on sift features. In Biometrics, Identity and Security (BIdS), 2009 International Conference on (pp. 1-8). IEEE.

Takacs, G., Chandrasekhar, V., Gelfand, N., Xiong, Y., Chen, W. C., Bismpigiannis, T., ... & Girod, B. (2008, October). Outdoors augmented reality on mobile phone using loxel-based visual feature organization. In Proceedings of the 1st ACM international conference on Multimedia

Han, H., & Jain, A. K. (2013, June). Tattoo based identification: Sketch to image matching. In Biometrics (ICB), 2013 Interna-tional Conference on (pp. 1-8). IEEE.

Ayers, B., & Boutell, M. (2007, June). Home interior classification using SIFT keypoint histograms. In Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on (pp. 1-6). IEEE.

Jinxia, L., & Yuehong, Q. (2011, August). Application of SIFT feature extraction algorithm on the image registration. In Electronic Measurement & Instruments (ICEMI), 2011 10th International Conference on (Vol. 3, pp. 177-180). IEEE.

Harris, C., & Stephens, M. (1988, August). A combined corner and edge detector. In Alvey vision conference (Vol. 15, p. 50).

Derpanis, K. G. (2004). The harris corner detector. York University.

Langlotz, T., Degendorfer, C., Mulloni, A., Schall, G., Reitmayr, G., & Schmalstieg, D. (2011). Robust detection and tracking of an-notations for outdoor augmented reality browsing. Computers & graphics, 35(4), 831-840.

Woods, R. Gonzales. R., 1992 Digital Image Processing, Boston: Addison-Wesley Publishing Company.

Richard, S., 2010 Computer vision: algorithms and applications. Heidelberg: Springer,.

7. APÉNDICE.

A lo largo del desarrollo de este trabajo de investigación se publico 1 articulo en un congreso nacional y 2 artículos en congresos internacionales, el primero de ellos fue publicado en el 10° Congreso Internacional de Ingeniería, con el titulo de: "Analysis and comparison of images using the SIFT algorithm with a pre-processing stage based on histograms with different lighting conditions".

El segundo fue publicado en XII International Conference on Mechatronics, Electronics and Automotive Engineering, con el titulo de "A Feature Extraction Using SIFT with a Preprocessing by Adding CLAHE Algorithm to Enhance Image Histograms."

También se encuentra en proceso de revisión un articulo enviado al Journal de Ingeniería e Investigación de la ciudad de Colombia con ISSN: 0120-5609.

A Feature Extraction Using SIFT with a Preprocessing by Adding CLAHE Algorithm to Enhance Image Histograms.

Palma Olvera R. D., Martínez Zerón E., Pedraza Ortega J. C., Ramos Arreguín J.M. and Gorrostieta Hurtado E.

Facultad de Informática Campus Juriquilla, Universidad Autónoma de Querétaro.

raulpolvera@gmail.com

***Abstract*—In this paper a novel method is proposed to improve the performance of the SIFT (Scale Invariant Feature Transformation) algorithm in adverse illumination conditions (in an outdoor environment at night), for this research it is proposed to work with CLAHE (Contrast Limited Adaptive Histogram Equalization), adding a preprocessing stage to the traditional methodology of the SIFT algorithm, this will be applied to the building to be found in the scene, i.e., the image pattern.**

A comparison with different illumination conditions (day, evening and night) will be held to know the response that will have the SIFT algorithm and to identify which moment the algorithm has a better performance.

Keywords-SIF; Pattern Recognition; Enhanced Histograms; CLAHE; Building Recognition.

I. INTRODUCTION.

Nowadays the pattern recognition and feature extraction algorithms are very important in the computer vision field, these algorithms are the basis of many applications today such as object recognition, image reconstruction, object tracking, 3D reconstruction, among others. One of the most popular algorithms used for feature extraction and pattern recognition is the SIFT algorithm (Scale Invariant Feature Transformation), which was introduced by Lowe [1], this algorithm is almost invariant to several factors that affect most of the algorithms in computer vision fields such as rotation, translation, occlusion, scale and slight illumination changes, for this is one of the most popular and widely used algorithms.

Some authors have addressed the issues of illumination problems that present computer vision algorithms, such as SIFT, by managing histograms, Tu [2] makes a comparison between SIFT and A-SIFT (a variation of the original SIFT algorithm proposed by Morel [3]), in which proposes an image enhancement by histogram equalization in blurred or images with slight changes of illumination, Palma [4] performed an analysis of the SIFT algorithm with preprocessed images by an enhancement based on histograms in different illumination conditions, showing good results in unfavorable lighting conditions. Tang [5] proposed the AH-SIFT, a variation of the SIFT algorithm using Augmented Histograms, this algorithm is based on augment the histogram of local image patch features with a set of circular means and variances.

This paper is structured as follows, Section II contains a summary of SIFT and CLAHE (Contrast Limited Adaptive Histogram Equalization) algorithms, in Section III methodology to improve the SIFT algorithm is presented, section IV shows the analysis and results and Section V concludes the our work.

II. BACKGROUND.

The SIFT algorithm presented by Lowe [1], is based on multiple scales and spaces, it focuses on transforming an image into a large collection of local features or points of interest [6], each of these is invariant to scale, rotation and a certain degree of illumination. This algorithm is divided into four sections [1], shown below.

A. Space Extreme Detection.

First, the images between two adjacent octaves are sampled by a factor of 2. Gaussian functions are used to smooth the image belonging to each of the octaves. Then the Gaussian pyramid is set, the Difference of Gaussian (DoG) pyramid is generated by the difference of Gaussian pyramid between two adjacent scales belonging to the same octave (Fig. 1), space pyramid of DoG [1] is generated considering:

$$L(x,y,\sigma) = G(x,y,\sigma) * I(x,y). \quad (1)$$

$$DoG(x,y,\sigma) = L(x,y,k\sigma) - L(x,y,\sigma). \quad (2)$$

Analysis and comparison of images using the SIFT algorithm with a preprocessing stage based on histogram with different lighting conditions.

Palma Olvera R. D. [1*], Martínez Zerón E. [1], Aceves Fernández M. A. [1], Tovar Arriaga S. [1], Ramos Arreguín J. M. [1] and Pedraza Ortega J. C. [1]
Facultad de Informática Campus Juriquilla, Universidad Autónoma de Querétaro.
*e-mail corresponding author:raulpolvera@gmail.com.

Abstract—This paper present an analysis and performance of the computer vision algorithm SIFT (Scale Invariant Feature Transformation), the algorithm will used for the recognition of an outdoor building[3], taking into consideration the possible variations due the changing luminosity and occlusion. An adjustment will be applied by techniques based on histograms before the SIFT algorithm[5].
A comparison was made based on the points of interest calculated in the recognition process using different reference values to the previously established in previous works[1][2], this way we can see the behavior of the SIFT algorithm in outdoor environments and the optimal values for this conditions.
Also to establish a reference between the distance of each point of interest or the amount these to correctly identified an object[4], because many false positives may occur due the similarity that may exist in the buildings or structures.

Índice de Términos— SIFT, Computer Vision, Feature Extraction, Histograms.

I. INTRODUCTION

Using pattern recognition algorithms in images has increased significantly in recent years, and are used for various applications and branches of science, such as: image reconstruction, automation imaging database, model-based recognition, object detection in video, visual data mining, texture recognition, machine vision robots, detection of symmetry, categorization of objects, among others[9].

To do this, one of the main requirements for obtaining these regions or points of interest that correspond to one or a previously analyzed images as a base model in different perspectives. One of the most important steps for the recognition of an image by different computer vision algorithms is the extraction of local descriptors, they have to be distinctive image, robust to occlusion and lighting changes. The idea of this is to detect regions in images that are covariant to different transformations that may arise in relation to occlusion, changes in illumination, scale and rotation, which would give us invariant descriptors for image analysis.

So would be the best descriptors to be drawn into a picture, because there are a lot of possible points to be an image descriptor considering different measures or properties of the image, such as intensity pixel, the color, texture, edges, among others, to exclude some of these properties is recommended to work with grayscale images[6].

The evaluation of the extracted descriptors is performed based on the recognition and pairing with another image of the same scene or object under different conditions, this can be done with either a single image or a database of the same scene to reduce further number of false positives.

II. PATTERN RECOGNITION ALGORITHMS: SIFT

The SIFT algorithm focuses on transforming an image into a large collection of local features[6], each of these is invariant to different transformations that could be the original image, scale, rotation, and partially invariant to illumination, for it is divided into four important phases algorithm.