



**ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO**  
**FACULTAD DE INFORMÁTICA Y ELECTRÓNICA**  
**ESCUELA DE INGENIERÍA ELECTRÓNICA EN CONTROL Y REDES**  
**INDUSTRIALES**

**“RECONOCIMIENTO Y CREACIÓN DEL MODELO**  
**FACIAL 3D MEDIANTE SISTEMA DE VIDEO APLICADO**  
**A LA SEGURIDAD USANDO INTELIGENCIA**  
**ARTIFICIAL”**

**TRABAJO DE TITULACIÓN: DISPOSITIVO TECNOLÓGICO**

**Para optar al grado académico de:**  
**INGENIERO EN ELECTRÓNICA, CONTROL Y REDES**  
**INDUSTRIALES**

**AUTORES: CARLOS SANTIAGO CABA COSTALES**  
**CAROLINE VIVIANA JARA CHÁVEZ**

**TUTOR: ING. PABLO EDUARDO LOZADA YÁNEZ**

**Riobamba – Ecuador**

**2018**

©2018, Carlos Santiago Caba Costales y Caroline Viviana Jara Chávez.

Se autoriza la reproducción total o parcial, con fines académicos, por cualquier medio o procedimiento, incluyendo a la cita bibliográfica del documento, siempre y cuando se reconozca el Derecho de Autor.

**ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO**  
**FACULTAD DE INFORMÁTICA Y ELECTRÓNICA**  
**ESCUELA DE INGENIERÍA ELECTRÓNICA EN CONTROL Y REDES**  
**INDUSTRIALES**

El Tribunal de Trabajo de Titulación certifica que: El trabajo de titulación: “RECONOCIMIENTO Y CREACIÓN DEL MODELO FACIAL 3D MEDIANTE SISTEMA DE VIDEO APLICADO A LA SEGURIDAD USANDO INTELIGENCIA ARTIFICIAL”, de responsabilidad de los señores Carlos Santiago Caba Costales y Caroline Viviana Jara Chávez, ha sido minuciosamente revisado por los Miembros del Tribunal de Titulación, quedando autorizada su presentación.

<b>NOMBRE</b>	<b>FIRMA</b>	<b>FECHA</b>
Dr. Julio Santillán C. <b>VICEDECANO DE LA FACULTAD DE INFORMÁTICA Y ELECTRÓNICA</b>	-----	-----
Ing. Freddy Chávez V. <b>DIRECTOR DE ESCUELA DE INGENIERÍA ELECTRÓNICA EN CONTROL Y REDES INDUSTRIALES</b>	-----	-----
Ing. Pablo Lozada Y. <b>DIRECTOR DEL TRABAJO DE TITULACIÓN</b>	-----	-----
Ing. Fernando Mejía. <b>MIEMBRO DEL TRIBUNAL</b>	-----	-----

Nosotros, CARLOS SANTIAGO CABA COSTALES y CAROLINE VIVIANA JARA CHÁVEZ, somos responsables de las ideas, doctrinas y resultados expuestos en este Trabajo de Titulación, y el patrimonio intelectual de la misma pertenece a la ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO.

Carlos Santiago Caba Costales

Caroline Viviana Jara Chávez

## **DEDICATORIA**

A Dios por iluminarme y darme fuerzas para culminar la carrera. A mis queridos padres por el apoyo incondicional durante mi carrera profesional y por estar pendiente de mí a cada momento. A mi familia, porque de una u otra forma, con su apoyo moral me han incentivado a seguir adelante, a lo largo de toda mi vida. Y a mis maestros que me brindaron sus conocimientos para poder culminar esta etapa de mi vida.

**Caroline Viviana**

Dedicado a mi Dios por darme salud, a mis padres y hermano quienes depositaron toda su confianza en mí, apoyándome incondicionalmente con esfuerzo y dedicación en cada paso de mi trayectoria.

Dedicado para todas las personas especiales que confiaron en mí, quienes con su granito de arena me ayudaron a motivar y hacer posible mi sueño anhelado.

**Carlos Santiago**

## **AGRADECIMIENTO**

A la Escuela Superior Politécnica de Chimborazo, a la Escuela de Ingeniería Electrónica Control y Redes Industriales, a los docentes, compañeros y amigos por todo el apoyo brindado durante la carrera.

A nuestros docentes que con paciencia y ahínco nos supieron impartir sus conocimientos en cada una de las asignaturas.

A los ingenieros Pablo Lozada y Fernando Mejía quien con su paciencia y conocimiento han sabido guiarnos con bases sólidas en la elaboración del presente trabajo.

Caroline Viviana y Carlos Santiago

## TABLA DE CONTENIDOS

ÍNDICE DE TABLAS-----	xi
ÍNDICE DE FIGURAS -----	xii
ÍNDICE DE GRÁFICOS -----	xvi
RESUMEN -----	xix
ABSTRACT-----	xx
INTRODUCCIÓN -----	1
<b>CAPÍTULO I</b>	
<b>1</b>	<b>MARCO TEÓRICO REFERENCIAL----- 8</b>
<b>1.1</b>	<b>Inteligencia Artificial ----- 8</b>
<b>1.1.1</b>	<b>Concepto ----- 8</b>
<b>1.1.2</b>	<b>Técnicas de inteligencia artificial----- 8</b>
1.1.2.1	<i>Redes Neuronales Artificiales (RNA)----- 8</i>
1.1.2.1.1	<i>Características de las Redes Neuronales----- 9</i>
1.1.2.2	<i>Algoritmos Genéticos----- 9</i>
1.1.2.3	<i>Lógica Difusa ----- 10</i>
1.1.2.3.1	<i>Pasos para la creación de un sistema difuso ----- 10</i>
1.1.2.3.2	<i>Tipos de sistemas de lógica difusa----- 11</i>
<b>1.2</b>	<b>Visión Artificial ----- 15</b>
<b>1.2.1</b>	<b>Concepto----- 15</b>
<b>1.2.2</b>	<b>Reconocimiento Facial----- 15</b>
1.2.2.1	<i>Algoritmos de visión artificial aplicados al reconocimiento facial.----- 16</i>
1.2.2.1.1	<i>Métodos holísticos ----- 16</i>
<b>1.2.3</b>	<b>Métodos basados en características locales----- 19</b>
1.2.3.1	<i>Estructura de un sistema de reconocimiento facial ----- 23</i>
1.2.3.1.1	<i>Imagen de entrada ----- 23</i>
1.2.3.1.2	<i>Detección del rostro ----- 23</i>
1.2.3.1.3	<i>Normalización ----- 26</i>
1.2.3.1.4	<i>Extracción de características----- 27</i>
1.2.3.1.5	<i>Reconocimiento (Comparación).----- 28</i>
1.2.3.2	<i>Resultados obtenidos ----- 29</i>

<b>1.2.4</b>	<b><i>Modelo Facial en 3D</i></b> -----	<b>29</b>
1.2.4.1	<i>Construcción 3DMM</i> -----	30
1.2.4.2	<i>Correspondencia densa</i> -----	30
1.2.4.3	<i>Alineación de similitud y modelado estadístico</i> -----	30
<b>1.3</b>	<b>Hardware</b> -----	<b>31</b>
<b>1.3.1</b>	<b><i>Controlador (Computador portátil TOSHIBA Satellite S855)</i></b> -----	<b>31</b>
<b>1.3.2</b>	<b><i>Web Cam Logitech C270</i></b> -----	<b>32</b>
<b>1.3.3</b>	<b><i>Arduino Mega 2560</i></b> -----	<b>33</b>
<b>1.3.4</b>	<b><i>Cerradura Eléctrica</i></b> -----	<b>35</b>
<b>1.3.5</b>	<b><i>Hub USB</i></b> -----	<b>35</b>
<b>1.3.6</b>	<b><i>Adaptador de corriente</i></b> -----	<b>36</b>
<b>1.3.7</b>	<b><i>Elementos electrónicos.</i></b> -----	<b>36</b>
1.3.7.1	<i>Resistencia</i> -----	36
1.3.7.2	<i>Optoacoplador</i> -----	36
1.3.7.3	<i>Relé</i> -----	37
1.3.7.4	<i>Diodo</i> -----	37
1.3.7.5	<i>Transistor</i> -----	38
<b>1.4</b>	<b>Software</b> -----	<b>38</b>
<b>1.4.1</b>	<b><i>Matlab</i></b> -----	<b>38</b>
1.4.1.1	<i>Características</i> -----	38
1.4.1.2	<i>Comandos y funciones de Matlab</i> -----	39
<b>1.4.2</b>	<b><i>MySQL server</i></b> -----	<b>40</b>
<b>CAPÍTULO II</b>		
<b>2</b>	<b>MARCO METODOLÓGICO</b> -----	<b>41</b>
<b>2.1</b>	<b>Proceso de elaboración del proyecto</b> -----	<b>41</b>
<b>2.1.1</b>	<b><i>Investigación de la información</i></b> -----	<b>41</b>
<b>2.1.2</b>	<b><i>Elección de los elementos a utilizar</i></b> -----	<b>41</b>
<b>2.1.3</b>	<b><i>Verificación de compatibilidad entre elementos.</i></b> -----	<b>41</b>
<b>2.1.4</b>	<b><i>Elaboración del software.</i></b> -----	<b>42</b>
<b>2.1.5</b>	<b><i>Descripción del lugar de aplicación.</i></b> -----	<b>42</b>
<b>2.1.6</b>	<b><i>Proceso de pruebas y planteamiento de resultados.</i></b> -----	<b>42</b>
<b>2.2</b>	<b>Estructura de la base de datos</b> -----	<b>42</b>
<b>2.3</b>	<b>Elaboración de la placa de conexión entre Matlab y los dispositivos.</b> -----	<b>43</b>
<b>2.3.1</b>	<b><i>Diseño de la placa</i></b> -----	<b>43</b>
<b>2.3.2</b>	<b><i>Principio de funcionamiento</i></b> -----	<b>44</b>
<b>2.4</b>	<b>Algoritmo de inteligencia artificial</b> -----	<b>46</b>

<b>2.4.1</b>	<b><i>Descripción del algoritmo de Lógica Difusa</i></b> -----	46
<b>2.4.2</b>	<b><i>Inclusión del algoritmo implementado en el sistema</i></b> -----	52
2.4.2.1	<i>Obtención de los datos de cada usuario</i> -----	52
2.4.2.2	<i>Definir cuál es la mayor coincidencia en relación a las demás</i> -----	56
2.4.2.3	<i>Determinar si el usuario seleccionado es correcto o no</i> -----	58
<b>2.5</b>	<b>Algoritmo de visión artificial</b> -----	59
<b>2.5.1</b>	<b><i>Captura de imagen</i></b> -----	59
<b>2.5.2</b>	<b><i>Carga del modelo genérico de rostro</i></b> -----	61
<b>2.5.3</b>	<b><i>Detección del rostro en la imagen</i></b> -----	62
<b>2.5.4</b>	<b><i>Normalización del rostro y sus partes</i></b> -----	64
<b>2.6</b>	<b>Descripción de las etapas del reconocimiento y reconstrucción 3D</b> -----	68
<b>2.6.1</b>	<b><i>Fase de adquisición de información</i></b> -----	69
2.6.1.1	<i>Descripción.</i> -----	69
2.6.1.2	<i>Comandos y funciones de Matlab</i> -----	71
<b>2.6.2</b>	<b><i>Fase de preprocesado de la información</i></b> -----	72
2.6.2.1	<i>Descripción</i> -----	72
2.6.2.2	<i>Comandos y Funciones de Matlab</i> -----	73
<b>2.6.3</b>	<b><i>Fase de ejecución del método seleccionado</i></b> -----	74
2.6.3.1	<i>Descripción</i> -----	74
2.6.3.2	<i>Comandos y Funciones de Matlab</i> -----	75
<b>2.6.4</b>	<b><i>Fase de comparación y resultados</i></b> -----	76
2.6.4.1	<i>Descripción</i> -----	76
<b>CAPÍTULO III</b>		
<b>3</b>	<b>MARCO DE RESULTADOS, ANÁLISIS Y DISCUSIÓN DE RESULTADOS</b> -----	<b>80</b>
<b>3.1</b>	<b>Análisis para determinar el método de detección facial (Objetivo 1)</b> -----	<b>80</b>
<b>3.1.1</b>	<b><i>Análisis en base al tiempo de respuesta en distintas posiciones</i></b> -----	<b>80</b>
<b>3.1.2</b>	<b><i>Análisis en base al número de intentos bajo ciertas condiciones</i></b> -----	<b>81</b>
<b>3.2</b>	<b>Diseño final del sistema (Objetivo 2 y 4)</b> -----	<b>82</b>
<b>3.3</b>	<b>Estructura de la base de datos implementada (Objetivo 3).</b> -----	<b>83</b>
<b>3.4</b>	<b>Conexión entre el software y los actuadores (Objetivo 5)</b> -----	<b>84</b>
<b>3.5</b>	<b>Análisis para determinar la calidad del proyecto (Objetivo 6)</b> -----	<b>86</b>
<b>3.5.1</b>	<b><i>Funcionalidad</i></b> -----	<b>86</b>
<b>3.5.2</b>	<b><i>Usabilidad</i></b> -----	<b>88</b>
<b>3.5.3</b>	<b><i>Eficiencia</i></b> -----	<b>91</b>
<b>3.6</b>	<b>Discusión de resultados.</b> -----	<b>94</b>

**CONCLUSIONES ----- 98**

**RECOMENDACIONES ----- 99**

**BIBLIOGRAFÍA**

**ANEXOS**

## ÍNDICE DE TABLAS

<b>Tabla 1-1:</b> Especificaciones de computador portátil toshiba satellite s855 -----	<b>32</b>
<b>Tabla 2-1:</b> Especificaciones de cámara web logitech c270 -----	<b>33</b>
<b>Tabla 3-1:</b> Especificaciones de arduino mega 2560 -----	<b>34</b>
<b>Tabla 1-2:</b> Comandos principales para la adquisición de información -----	<b>71</b>
<b>Tabla 2-2:</b> Comandos auxiliares para la adquisición de información -----	<b>71</b>
<b>Tabla 3-2:</b> Tamaños de imágenes normalizadas -----	<b>72</b>
<b>Tabla 4-2:</b> Tabla de usuarios creada en la base de datos -----	<b>73</b>
<b>Tabla 5-2:</b> Comandos principales para la fase de preprocesado de la imagen -----	<b>73</b>
<b>Tabla 6-2:</b> Funciones para la fase de preprocesado de la imagen -----	<b>74</b>
<b>Tabla 7-2:</b> Comandos principales para la fase de ejecución del método seleccionado -----	<b>75</b>
<b>Tabla 8-2:</b> Funciones para la fase de ejecución del método seleccionado -----	<b>75</b>
<b>Tabla 9-2:</b> Tabla de admitidos creada en la base de datos -----	<b>77</b>
<b>Tabla 10-2:</b> Tabla de rechazados creada en la base de datos -----	<b>77</b>
<b>Tabla 1-3:</b> Tiempo de respuesta de los dos métodos. -----	<b>80</b>
<b>Tabla 2-3:</b> Intentos de detección bajo ciertas condiciones -----	<b>81</b>
<b>Tabla 3-3:</b> Datos obtenidos del funcionamiento del sistema -----	<b>86</b>
<b>Tabla 4-3:</b> Resultados de pruebas de funcionamiento -----	<b>87</b>
<b>Tabla 5-3:</b> Porcentaje de ayuda en proceso de registro -----	<b>88</b>
<b>Tabla 6-3:</b> Porcentaje de ayuda en manejo del sistema -----	<b>89</b>
<b>Tabla 7-3:</b> Porcentaje de ayuda en petición de acceso -----	<b>90</b>
<b>Tabla 8-3:</b> Rangos tiempo de espera -----	<b>91</b>
<b>Tabla 9-3:</b> Porcentaje de tiempo de respuesta -----	<b>92</b>
<b>Tabla 10-3:</b> Porcentaje de tiempo de respuesta -----	<b>92</b>
<b>Tabla 11-3:</b> Porcentaje de tiempo de respuesta -----	<b>93</b>
<b>Tabla 12-3:</b> Resumen de usabilidad del sistema. -----	<b>95</b>
<b>Tabla 13-3:</b> Resumen de eficiencia del sistema. -----	<b>96</b>

## ÍNDICE DE FIGURAS

<b>Figura 1:</b> Funcionamiento del sistema propuesto -----	<b>6</b>
<b>Figura 1-1:</b> Sistema difuso mamdani proceso general -----	<b>11</b>
<b>Figura 2-1:</b> Regla de la base de conocimientos -----	<b>12</b>
<b>Figura 3-1:</b> Sistema difuso mamdani proceso detallado -----	<b>13</b>
<b>Figura 4-1:</b> Sistema difuso sugeno procesamiento general -----	<b>13</b>
<b>Figura 5-1:</b> Regla de base de conocimiento de sistema sugeno -----	<b>14</b>
<b>Figura 6-1:</b> Regla de base de conocimiento de sistema sugeno -----	<b>14</b>
<b>Figura 7-1:</b> Algoritmos de reconocimiento facial -----	<b>16</b>
<b>Figura 8-1:</b> Imagen nxn transformada a vector $n^2 \times 1$ -----	<b>17</b>
<b>Figura 9-1:</b> Ejemplo de seis clases usando LDA -----	<b>18</b>
<b>Figura 10-1:</b> Separación de puntos de entrada mediante línea discontinua -----	<b>19</b>
<b>Figura 11-1:</b> Método AAM -----	<b>20</b>
<b>Figura 12-1:</b> Método EBGm -----	<b>21</b>
<b>Figura 13-1:</b> Diagrama general del reconocimiento facial -----	<b>23</b>
<b>Figura 14-1:</b> Modelo unificado para la detección del rostro -----	<b>25</b>
<b>Figura 15-1:</b> Modelo unificado para la detección del rostro -----	<b>25</b>
<b>Figura 16-1:</b> Detección de rasgos faciales -----	<b>27</b>
<b>Figura 17-1:</b> Coincidencia de un modelo morphable con una sola imagen -----	<b>30</b>
<b>Figura 18-1:</b> Combinación automática a un modelo morfálico -----	<b>31</b>
<b>Figura 19-1:</b> Computador portatil toshiba satellite S855. -----	<b>31</b>
<b>Figura 20-1:</b> Cámara web logitech c270. -----	<b>32</b>

<b>Figura 21-1:</b> Logo arduino mega 2560. -----	<b>33</b>
<b>Figura 22-1:</b> Cerradura eléctrica. -----	<b>35</b>
<b>Figura 23-1:</b> Conector hub usb -----	<b>35</b>
<b>Figura 24-1:</b> Adaptador de corriente -----	<b>36</b>
<b>Figura 25-1:</b> Resistencia -----	<b>36</b>
<b>Figura 26-1:</b> Optoacoplador -----	<b>37</b>
<b>Figura 27-1:</b> Relé -----	<b>37</b>
<b>Figura 28-1:</b> Diodo -----	<b>37</b>
<b>Figura 29-1:</b> Transistor -----	<b>38</b>
<b>Figura 30-1:</b> Logo matlab -----	<b>38</b>
<b>Figura 31-1:</b> Logo MySQL -----	<b>40</b>
<b>Figura 1-2:</b> Tablas de la base de datos del proyecto -----	<b>43</b>
<b>Figura 2-2:</b> Diseño de pistas del circuito -----	<b>44</b>
<b>Figura 3-2:</b> Circuito de control de actuadores -----	<b>45</b>
<b>Figura 4-2:</b> Estructura de algoritmo de lógica difusa -----	<b>46</b>
<b>Figura 5-2:</b> Modelo de lógica difusa -----	<b>47</b>
<b>Figura 6-2:</b> Variable PD -----	<b>48</b>
<b>Figura 7-2:</b> Variable NA -----	<b>49</b>
<b>Figura 8-2:</b> Variable C -----	<b>49</b>
<b>Figura 9-2:</b> Reglas fuzzy del sistema -----	<b>50</b>
<b>Figura 10-2:</b> Resultados fuzzy forma dinámica -----	<b>51</b>
<b>Figura 11-2:</b> Resultados fuzzy forma 3D -----	<b>51</b>
<b>Figura 12-2:</b> Algoritmo de inteligencia artificial -----	<b>52</b>

<b>Figura 13-2:</b> Código de obtención banco de filtros de los usuarios -----	<b>52</b>
<b>Figura 14-2:</b> Rostro de frente del usuario-----	<b>53</b>
<b>Figura 15-2:</b> Vectores característicos rostro-----	<b>53</b>
<b>Figura 16-2:</b> Vectores característicos nariz de frente -----	<b>54</b>
<b>Figura 17-2:</b> Vectores característicos boca de frente-----	<b>54</b>
<b>Figura 18-2:</b> Vectores característicos ojos de frente -----	<b>54</b>
<b>Figura 19-2:</b> Rostro de perfil del usuario -----	<b>55</b>
<b>Figura 20-2:</b> Vectores característicos rostro perfil -----	<b>55</b>
<b>Figura 21-2:</b> Vectores característicos nariz de perfil -----	<b>55</b>
<b>Figura 22-2:</b> Vectores característicos boca de perfil -----	<b>56</b>
<b>Figura 23-2:</b> Vectores característicos ojos de perfil-----	<b>56</b>
<b>Figura 24-2:</b> Porcentajes de similitud con cuatro usuarios registrado en la base de datos (frente) -----	<b>57</b>
<b>Figura 25-2:</b> Porcentajes de similitud con cuatro usuarios registrado en la base de datos (perfil) -----	<b>57</b>
<b>Figura 26-2:</b> Código que permite decidir si es usuario registrado o no -----	<b>58</b>
<b>Figura 27-2:</b> Algoritmo de visión artificial-----	<b>59</b>
<b>Figura 28-2:</b> Código de captura de imagen en matlab -----	<b>60</b>
<b>Figura 29-2:</b> Imagen de entrada de frente del usuario-----	<b>60</b>
<b>Figura 30-2:</b> Imagen de entrada de perfil del usuario -----	<b>61</b>
<b>Figura 31-2:</b> Código de carga del modelo en matlab-----	<b>61</b>
<b>Figura 32-2:</b> Modelos empleados para el reconocimiento de frente-----	<b>62</b>
<b>Figura 33-2:</b> Modelos empleados para el reconocimiento de perfil -----	<b>62</b>

<b>Figura 34-2:</b> Código de la detección de rostro en matlab-----	<b>62</b>
<b>Figura 35-2:</b> Detección del rostro en la imagen de frente -----	<b>63</b>
<b>Figura 36-2:</b> Detección del rostro en la imagen de perfil-----	<b>64</b>
<b>Figura 37-2:</b> Código de normalización y recorte de rostro y sus partes-----	<b>64</b>
<b>Figura 38-2:</b> Recorte del rostro de frente -----	<b>65</b>
<b>Figura 39-2:</b> Normalización del rostro de frente-----	<b>65</b>
<b>Figura 40-2:</b> Recorte y normalización boca-----	<b>66</b>
<b>Figura 41-2:</b> Recorte y normalización nariz-----	<b>66</b>
<b>Figura 42-2:</b> Recorte y normalización de ojos -----	<b>66</b>
<b>Figura 43-2:</b> Recorte del rostro de perfil-----	<b>67</b>
<b>Figura 44-2:</b> Normalización del rostro de perfil -----	<b>67</b>
<b>Figura 45-2:</b> Recorte y normalización boca -----	<b>67</b>
<b>Figura 46-2:</b> Recorte y normalización nariz-----	<b>68</b>
<b>Figura 47-2:</b> Recorte y normalización de ojos -----	<b>68</b>
<b>Figura 48-2:</b> Etapas del reconocimiento y creación del modelo facial 3D -----	<b>69</b>
<b>Figura 49-2:</b> Adquisición de imágenes del usuario que solicita acceso-----	<b>70</b>
<b>Figura 50-2:</b> Acciones que se toman cuando un usuario es aceptado o no -----	<b>79</b>
<b>Figura 51-2:</b> Modelo facial 3D-----	<b>79</b>
<b>Figura 1-3:</b> Diseño del sistema de reconocimiento facial-----	<b>83</b>
<b>Figura 2-3:</b> Diseño del sistema de reconocimiento facial-----	<b>84</b>
<b>Figura 3-3:</b> Circuito de control de actuadores-----	<b>85</b>

## ÍNDICE DE GRÁFICOS

<b>Gráfico 1-3:</b> Porcentajes de los métodos en tiempo de respuesta -----	<b>81</b>
<b>Gráfico 2-3:</b> Porcentajes de los métodos bajo ciertas condiciones -----	<b>82</b>
<b>Gráfico 3-3:</b> Porcentaje de funcionamiento del sistema -----	<b>88</b>
<b>Gráfico 4-3:</b> Porcentaje de ayuda en proceso de registro -----	<b>89</b>
<b>Gráfico 5-3:</b> Porcentaje de ayuda en manejo del sistema -----	<b>90</b>
<b>Gráfico 6-3:</b> Porcentaje de ayuda en petición de acceso -----	<b>91</b>
<b>Gráfico 7-3:</b> Porcentaje de respuesta para el registro al sistema -----	<b>92</b>
<b>Gráfico 8-3:</b> Porcentaje de respuesta al reconocimiento facial del sistema-----	<b>93</b>
<b>Gráfico 9-3:</b> Porcentaje de respuesta del modelo facial 3d del sistema -----	<b>94</b>
<b>Gráfico 10-3:</b> Funcionalidad del sistema -----	<b>95</b>
<b>Gráfico 11-3:</b> Usabilidad del sistema -----	<b>96</b>
<b>Gráfico 12-3:</b> Eficiencia del sistema -----	<b>97</b>

## ÍNDICE DE ABREVIATURAS

<b>EBGM</b>	Elastic Bunch Graph Model
<b>PCA</b>	Análisis de Componentes Principales
<b>LDA</b>	Análisis Lineal Discriminante
<b>ICA</b>	Análisis de Componentes Independientes
<b>SVM</b>	Máquinas de vectores de Soporte
<b>AAM</b>	Modelo de Apariencia Activa
<b>HMM</b>	Modelo Oculto de Markov
<b>LBP</b>	Patrones Binarios Locales
<b>3DMM</b>	Modelos 3D Morphable
<b>SQL</b>	Structured Query Language

## ÍNDICE DE ANEXOS

- Anexo A:** Conexión Matlab y MySQL
- Anexo B:** Conexión Arduino Matlab
- Anexo C:** Instalación paquetes de Mathworks
- Anexo D:** Manual de usuario

## **RESUMEN**

En el presente trabajo, se diseña e implementa un sistema de reconocimiento y creación del modelo facial 3D que simula cualquier tipo de ingreso a un determinado lugar. El sistema cuenta con dos etapas: la etapa de registro de usuarios y la etapa de ejecución del programa principal en donde se ejecuta el reconocimiento y creación del modelo. Se usa el método de detección en base a un modelo genérico del rostro. Para la validación de ingreso, se utiliza la técnica de inteligencia artificial empleada es lógica difusa, cuyo controlador difuso es de tipo Mamdani y con defuzzificador centroide; este se utiliza para validar la similitud entre el usuario activo y la información almacenada en la base de datos. Para la elaboración de vectores característicos de los usuarios se emplean filtros de Gabor y para la comparación se aplica la distancia euclidiana. Para tener más precisión al momento de reconocer una persona se usa como entrada varias imágenes como: imagen del rostro total, ojos, boca, nariz, tanto de frente como de perfil. Se determina que el sistema elimina la restricción de que la imagen de entrada tenga luz, fondo y postura exactos que presentan otros sistemas. Se concluye que, al utilizar un algoritmo de inteligencia artificial como medio para la selección del usuario al cual se permite el acceso, se asegura evitar un error de identificación, debido a que muchas veces un rasgo en particular de una persona suele ser idéntico o al menos casi parecido al de otra, pero no su rostro total. Si llegara a elaborarse algún otro algoritmo de reconocimiento es recomendable tener en cuenta el tiempo de respuesta del mismo debido a que al tratarse de seguridad automatizada los aspectos importantes son la confiabilidad y eficiencia del sistema.

**PALABRAS CLAVES:** <TECNOLOGÍA Y CIENCIAS DE LA INGENIERÍA>, <INTELIGENCIA ARTIFICIAL>, <SISTEMA DE RECONOCIMIENTO FACIAL>, <CONTROL DE ACCESO>, <MODELO FACIAL 3D>, <DETECCIÓN FACIAL>, <LÓGICA DIFUSA>, <FILTROS GABOR>.

## **ABSTRACT**

With the present work designed and implemented a system of recognition and creation of the 3D facial model that simulates any type of input to a certain place. The system involves two stages: the user registration stage, and the execution stage of the main program in which the recognition and creation of the model is carried out. It used the detection method using a generic model of the face, besides the method of artificial intelligence was fuzzy logic, whose fuzzy controller is the Mamdani type and with a defuzzifier that was used to validate the similarity between the active user, and the information stored in the database. For the production of characteristic vectors of the users, applied the Gabor filters and for comparison used the Euler distance. In order to have more precision when recognizing a person used as input eight images such as: full face, eyes, mouth, and nose in profile and front view. It was obtained that the system eliminated the restriction for the input image has light, background, and exact position that other systems require. It is concluded that, by using an artificial intelligence algorithm as a means to select the user to whom access is allowed. On the other hand, it ensures to avoid and identification error because many times a particular trait of a person is usually identical or almost similar to another, but not his total face. If any other recognition algorithm were developed, it is advisable to take into account the response time of the algorithm; since when dealing with automated security, the important aspects are reliability and response time.

**KEYWORDS:** <ENGINEERING TECHNOLOGY AND SCIENCE>, <ARTIFICIAL INTELLIGENCE>, <FACIAL RECOGNITION SYSTEM>, <ACCESS CONTROL>, <3D FACIAL MODEL>, <FACIAL DETECTION>, <DIFFUSE LOGIC>, <GABOR FILTERS>.

## INTRODUCCIÓN

El reconocimiento facial es un tema predominante en la ciencia de la computación específicamente en el área de la Inteligencia Artificial; se enfoca en el estudio e implementación de sistemas biométricos que se basan en métodos automáticos para reconocer a personas sobre la base del análisis de sus características fisiológicas o de comportamiento.

El avance de la tecnología ha permitido la automatización y perfeccionamiento de los procesos de reconocimiento biométrico, en especial los relacionados con la seguridad, por lo que han ido surgiendo nuevos avances tecnológicos como las cámaras digitales, dispositivos móviles, entre otros, pudiéndose encontrar estos sistemas de reconocimiento facial en lugares donde se necesite tener un control de acceso a una cierta área, o en sistemas de seguridad para acceder a lugares.

En la actualidad la seguridad es un tema que tiene gran importancia a nivel mundial por tal motivo se desarrolla el trabajo de titulación “Reconocimiento y creación del modelo facial 3d mediante sistema de video aplicado a la seguridad usando inteligencia artificial”, el cual consta de dos partes: la primera consiste en el reconocimiento facial para el control de acceso y la segunda es la creación de un modelo facial 3D para identificar posibles intrusos en el lugar de acceso reforzando la parte de seguridad del sistema.

El reconocimiento facial como tal es una gran herramienta para el control de seguridad en accesos pero se decide implementar un algoritmo de inteligencia artificial, más específicamente un algoritmo de lógica difusa para complementar su función y minimizar así las variables que representan un problema para el reconocimiento como son la iluminación, la posición del usuario, color de piel, etc.

El trabajo se divide en tres capítulos, en el primer capítulo se describe las características técnicas del controlador, cámaras, microcontroladores, dispositivos de entrada y salida y demás hardware que se utiliza, también se describe los programas en los que se desarrolla el sistema, en este caso Matlab y MySQL, por último se da a conocer la definición de visión e inteligencia artificial, así como de sus técnicas y algoritmos que son reconocimiento EBGMM, creación de modelo 3DMM y lógica difusa.

En el segundo capítulo se indica el proceso que se sigue para la elaboración del sistema, la estructuración del motor de base de datos que se utiliza, como se elabora la placa que se implementa para adaptar los dispositivos de entrada y salida con el controlador, se describe

paso a paso la técnica de lógica difusa que se implementa, se detalla cómo se estructura y aplica el algoritmo de reconocimiento EBGM y de creación del modelo facial 3DMM.

En el tercer capítulo se indica los resultados obtenidos en base a los objetivos planteados. Por último se muestra las conclusiones y recomendaciones del trabajo.

## **ANTECEDENTES**

La tecnología con el crecimiento abundante que ha venido mostrando los últimos años, se está incluyendo mucho en nuestra vida diaria, hoy en día dependemos casi para todo de la tecnología y la seguridad no es la excepción.

Al hablar de seguridad nos referimos a nuestros hogares o puestos de trabajo y nuestro tema de investigación se centra exclusivamente en el acceso a dichos lugares. En esta misma área de investigación se cuenta con múltiples opciones, se puede mencionar accesos por características biométricas, claves personalizadas, lectores de tarjetas personales, incluso reconocimiento facial u ocular; el inconveniente que poseen todas las formas de acceso es no interactuar directamente con el usuario y la posibilidad de ser fácilmente saboteadas.

Se puede ver la evolución del reconocimiento facial desde tres perspectivas que se describen a continuación:

### **Nivel internacional:**

A nivel mundial en los países desarrollados los sistemas de reconocimiento facial ha evolucionado considerablemente en diferentes aplicaciones desde identificación ciudadana a gran escala por ejemplo, en los aeropuertos se han empezado a introducir diversos sistemas de reconocimiento facial, de iris y de huella dactilar con el objetivo de que los pasajeros puedan acceder a cualquier zona del aeropuerto, aplicaciones de vigilancia, aplicaciones relacionadas a la ley para la identificación de delincuentes entre otras.

### **Nivel regional:**

En los sistemas bancarios de Latinoamérica, se están implementando sistemas biométricos de seguridad por completo, los cuales se harán en dos fases distintas: en la primera, se instalará en los servicios de banca móvil de los clientes de más peso de la entidad ecuatoriana.

Posteriormente, se extenderá al resto de la cartera de usuarios del banco, que tiene presencia en las 24 provincias del país a través de 158 oficinas y 4.818 Bancos de Barrio. (Tapia, 2015).

### **Nivel local:**

El reconocimiento facial a nivel local aún no ha sido definido como un estándar en los diversos campos, siendo utilizada esta tecnología solo en el plano individual, sin embargo en algunos edificios ya se ha implementado sistemas de reconocimiento facial para poder controlar el personal o ya sea por motivos de seguridad. (Martínez, 2016, p. 12).

El problema del sistema que actualmente domina en el mercado en cuanto a reconocimiento facial aplicado a la seguridad se refiere, son las estrictas condiciones que se presenta a la hora de capturar la imagen del usuario y es por eso que se propone eliminar esas restricciones logrando así mayor conformidad y a la vez sencillez a la hora de asegurar algún lugar deseado.

## **FORMULACIÓN DEL PROBLEMA**

¿Es posible implementar un sistema de reconocimiento y modelado facial a través de un sistema de seguridad mediante cámaras de video el cual nos proporcione una mayor seguridad aplicando inteligencia artificial?

## **SISTEMATIZACIÓN DEL PROBLEMA**

¿Con el análisis de la información se puede elaborar un sistema de reconocimiento dentro de la visión artificial?

¿Se puede implementar inteligencia artificial en el reconocimiento facial?

¿Se puede gestionar la creación y ejecución de una base de datos que nos ayudará en el proceso de reconocimiento facial?

¿Se puede relacionar el reconocimiento facial con un actuador que controle físicamente el acceso de algún lugar?

¿Se puede usar inteligencia artificial en la creación del modelo facial 3D?

¿Será posible validar y verificar que nuestro sistema funciona correctamente?

## **JUSTIFICACIÓN**

### **JUSTIFICACIÓN TEÓRICA**

El sistema de reconocimiento facial es una aplicación dirigida por ordenador que identifica automáticamente a una persona en una imagen digital. Esto es posible mediante un análisis de las características faciales del sujeto extraídas de la imagen o de un fotograma clave de una fuente de video, y comparándolas con una base de datos. (Blázquez, 2013, p. 6).

Existen tres enfoques relacionados al tema de reconocimiento facial que son:

1. Enfoque geométrico basado en los rasgos del rostro.
2. Enfoque fotométrico basado en lo visual.
3. Enfoque basado en la combinación de los dos anteriores.

El enfoque con el que se va a trabajar será el combinado ya que el sistema reconoce rasgos faciales ojos nariz boca (geométrico), además de aportar información de la cara (fotométrico) y para el desarrollo del mismo existen varios algoritmos de reconocimiento de los cuales el algoritmo que más se adapta a nuestras necesidades es el algoritmo EBGM (Elastic Bunch Graph) porque una de las características del algoritmo EBGM es que trabaja con algunos puntos de interés del rostro y no con la cara en su totalidad, también otra característica es que no necesariamente se necesita una captura de imagen frontal de la persona que se va a reconocer sino que puede tener como entrada varias imágenes en distintas posiciones, en nuestro trabajo se toma como imagen de entrada una en posición frontal y otra de perfil. (Blázquez, 2013, p. 6).

En cuanto a la creación del modelo facial 3D se implementa el método de síntesis Morphable Model For The Synthesis Of 3D (3DMM) debido a que este método tiene como principal atributo que necesita de entrada un par de imágenes ortogonales del rostro de la persona que se piensa recrear, y como previamente hablamos de que un requisito del reconocimiento facial es que se tenga múltiples capturas del rostro de la persona se puede aplicar la misma información también para la parte de creación del modelo facial 3D.

Adicionalmente, con el fin de optimizar el proceso del modelado facial, se utiliza técnicas de Inteligencia Artificial que permitirá discriminar los puntos de interés para seleccionar aquellos que permitan la creación del modelo facial 3D con un elevado índice de fiabilidad.

La técnica de Inteligencia Artificial que se utilice será el que mejor tiempo de respuesta nos dé al momento de comprobar su funcionamiento ya sea este una red neuronal, lógica difusa, algoritmo genético.

El proceso de construcción del modelo genérico para 3DMM depende de los puntos de ajuste que se vayan a utilizar, de manera que la malla 3D resultante está conformada principalmente por los triángulos que forman estos puntos. Por otro lado, para el proceso de ajuste de la malla facial genérica (Generic Face Model, GFM) a una imagen con un rostro específico se ha optado por la utilización de 2 ejes ópticos en forma que se tengan imágenes ortogonales de la persona a la cual deseamos crear el modelo facial 3D. (Combarro, y otros, 2015)

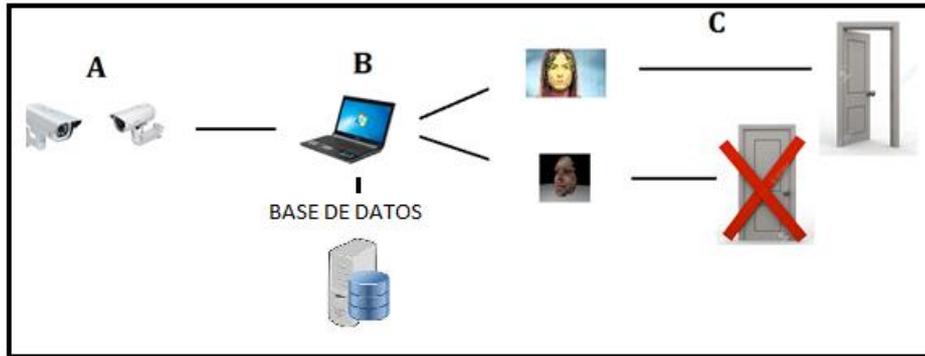
## **JUSTIFICACIÓN APLICATIVA**

La realización de este trabajo se basa en que existe una gran acogida actualmente de los sistemas de video vigilancia, y que la necesidad de tener seguridad en los lugares de trabajo o en los lugares donde residen las personas está incrementándose.

De forma práctica el sistema que se propone utiliza procesamiento de imágenes que puede ser adaptado a todos los sistemas de video vigilancia que existen en la actualidad siendo así una gran ventaja para los usuarios ya que no resultará ser un gasto significativo adicional para aquellas personas que desearan implementar este tipo de seguridad.

Además otro punto a favor que posee nuestro sistema es ser imperceptible ante la gente es decir ante cualquier persona parecería un simple sistema de grabación, pero no consideran que en si el sistema de reconocimiento y creación del modelo facial 3D esta internamente configurado para controlar el acceso al lugar donde este implementado.

En cuanto a su funcionamiento se pretende que el sistema trabaje en 3 fases principales que se las describe en la Figura 1.



**FIGURA 1:** Funcionamiento del sistema propuesto  
**Realizado por:** JARA, Caroline; CABA, Carlos, 2018

**FASE A:** En esta etapa se realiza la adquisición de datos por medio de las cámaras que estarán ubicadas de forma adecuada dependiendo del lugar de instalación.

**FASE B:** En esta etapa se realiza el procesamiento de la información con el método elegido y adicional se optimizará sus resultados aplicando inteligencia artificial.

Esta información se comparará con la almacenada en una base de datos de las posibles personas que vayan a tener acceso al lugar, la base de datos la desarrollaremos en el software MySQL server porque es un sistema de libre distribución y de código abierto

**FASE C:** Esta etapa es dependiente de los resultados de la fase B, por un lado se muestra el resultado de la persona que se ha identificado si en caso coincide con la base de datos y se procede a abrir la puerta para permitir su acceso, por otra parte se realiza una creación del modelo facial 3D si no existiese dicha persona en la base de datos, también se almacena la información de la persona y se bloqueará la entrada al lugar.

Por lo tanto esta investigación provee las pautas necesarias para proseguir en el avance del conocimiento científico y tecnológico de la seguridad basada en la tecnología. Específicamente en el proceso de reconocimiento y reconstrucción facial, para el control de acceso a diversos lugares. Conforme a lo estipulado en el Plan Nacional del Buen Vivir del Ecuador. Concretamente lo definido dentro del objetivo 10, literal a) del lineamiento 10.2, que expresa: “Articular la investigación científica, tecnológica y la educación superior con el sector productivo, para una mejora constante de la productividad y competitividad sistémica, en el marco de las necesidades actuales y futuras del sector productivo y el desarrollo de nuevos conocimientos. (Desarrollo, 2016).

## **OBJETIVOS**

### **OBJETIVO GENERAL**

- Implementar un sistema de seguridad mediante cámaras de video aplicando un proceso de reconocimiento y modelamiento facial en 3D utilizando métodos de inteligencia artificial.

### **OBJETIVOS ESPECÍFICOS**

- Analizar la información del reconocimiento y modelo facial 3D dentro de la visión artificial.
- Diseñar un sistema de reconocimiento facial aplicando un algoritmo de inteligencia artificial.
- Diseñar una base de datos que reúna información específica de las características de los usuarios admitidos y rechazados mediante el software MySQL server.
- Realizar la creación del modelo facial en 3D para la identificación de posibles intrusos que no estén registrados en la base de datos.
- Realizar la conexión física con un actuador que permita o restrinja el acceso de acuerdo a los resultados del reconocimiento facial.
- Realizar la validación y verificación del sistema automático.

# CAPÍTULO I

## 1 MARCO TEÓRICO REFERENCIAL

### 1.1 Inteligencia Artificial

#### 1.1.1 *Concepto*

Se refiere a máquinas que “piensan como seres humanos”; en otras palabras, que realizan tareas como: razonar, planificar, aprender y entender el lenguaje. Nadie espera que en la actualidad, o en un futuro cercano, una tecnología pueda igualar la inteligencia humana, pero la IA tiene grandes implicaciones en la forma en que vivimos nuestras vidas. El cerebro de la inteligencia artificial es una tecnología llamada aprendizaje automático, que está diseñada para hacer nuestro trabajo más fácil y productivo. (salesforce.com, 2000-2017)

#### 1.1.2 *Técnicas de inteligencia artificial*

Una técnica de inteligencia artificial es un método que utiliza conocimiento expresado, de tal forma que represente las generalizaciones, que sea comprendido por las personas que lo proporcionan, que pueda modificarse fácilmente para corregir errores y reflejar los cambios en el mundo y en nuestra visión del mundo, que pueda usarse en gran cantidad de situaciones aun cuando no sea totalmente preciso o completo y que pueda usarse para ayudar a superar su propio volumen, ayudando a acotar el rango de posibilidades que normalmente deben ser consideradas. (Molina J., Torres C., Restrepo C., 2008, p. 136)

##### 1.1.2.1 *Redes Neuronales Artificiales (RNA)*

Son un paradigma de aprendizaje y procesamiento automático inspirado en la forma en que funciona el sistema nervioso de los animales. Se trata de un sistema de interconexión de neuronas en una red que colabora para producir un estímulo de salida.

Las RNA son modelos que intentan reproducir el comportamiento del cerebro. Como tal modelo, realiza una simplificación, averiguando cuáles son los elementos relevantes del sistema, bien porque la cantidad de información de que se dispone es excesiva o bien porque es redundante. Una elección adecuada de sus características, más una estructura conveniente, es el

procedimiento convencional utilizado para construir redes capaces de realizar determinada tarea.

#### 1.1.2.1.1 *Características de las Redes Neuronales*

Existen cuatro aspectos que caracterizan una red neuronal: su topología, el mecanismo de aprendizaje, tipo de asociación realizada entre la información de entrada y salida, y la forma de representación de estas informaciones.

**Redes Monocapa:** Se establecen conexiones laterales, cruzadas o autor recurrentes entre las neuronas que pertenecen a la única capa que constituye la red. Se utilizan en tareas relacionadas con lo que se conoce como auto asociación; por ejemplo, para generar informaciones de entrada que se presentan a la red incompleta o distorsionada.

**Redes Multicapa:** Son aquellas que disponen de conjuntos de neuronas agrupadas en varios niveles o capas. Una forma de distinguir la capa a la cual pertenece la neurona, consiste en fijarse en el origen de las señales que recibe a la entrada y el destino de la señal de salida. Según el tipo de conexión, como se vio previamente, se distinguen las redes feedforward, y las redes feedforward/feedback.

**Topología de las Redes Neuronales:** La arquitectura de las redes neuronales consiste en la organización y disposición de las neuronas formando capas más o menos alejadas de la entrada y salida de la red. En este sentido, los parámetros fundamentales de la red son: el número de capas, el número de neuronas por capa, el grado de conectividad y el tipo de conexiones entre neuronas.

**Mecanismo de Aprendizaje:** El aprendizaje es el proceso por el cual una red neuronal modifica sus pesos en respuesta a una información de entrada. Los cambios que se producen durante el proceso de aprendizaje se reducen a la destrucción, modificación y creación de conexiones entre las neuronas. La creación de una nueva conexión implica que el peso de la misma pasa a tener un valor distinto de cero, una conexión se destruye cuando su peso pasa a ser cero. (EcuRed, Marzo 2018).

#### 1.1.2.2 *Algoritmos Genéticos*

Los Algoritmos Genéticos son métodos adaptativos que pueden usarse para resolver problemas de búsqueda y optimización. Están basados en el proceso genético de los organismos vivos. A lo largo de las generaciones, las poblaciones evolucionan en la naturaleza de acorde con los

principios de la selección natural y la supervivencia de los más fuertes, postulados por Darwin (1859).

Por imitación de este proceso, los Algoritmos Genéticos son capaces de ir creando soluciones para problemas del mundo real. La evolución de dichas soluciones hacia valores óptimos del problema depende en buena medida de una adecuada codificación de las mismas.

### 1.1.2.3 *Lógica Difusa*

La lógica difusa puede usarse para explicar el mundo en el que vivimos ya que sigue el comportamiento humano de razonar sacando conclusiones de hechos observados.

La lógica difusa está definida como un sistema matemático que modela funciones no lineales convirtiendo unas entradas en salidas de acuerdo a los planteamientos lógicos que usa el razonamiento aproximado pudiendo ver a un sistema de información como un sistema difuso, esto quiere decir que se hace uso de los principios de la lógica difusa para identificar la base del conocimiento, variables lingüísticas, funciones de membresía de un sistema y modelarlo matemáticamente esto ayuda al diseño y modelado de sistemas de software con interacción humana. (García, 2006, p. 11).

Medina & Paniagua (2007) señalan que una de las principales ventajas de la lógica difusa es que permite definir e incluir en un análisis conceptos o variables, aun cuando no estén formulados de forma precisa. (Santana, 2013, p. 500).

#### 1.1.2.3.1 *Pasos para la creación de un sistema difuso*

Proceso de parametrización que define a las variables de entrada y salida del sistema funciones de pertinencia cuyos valores son dados generalmente por opiniones de expertos que se distribuye dentro del posible rango que puede tomar la variable, estas funciones de pertinencia dependen del contexto y la aplicación que se vaya a tratar algunas de las funciones de pertinencia típica son: Función triangular; función gama; función s; función gaussiana; función trapezoidal; función pseudo-exponencial; función trapecio extendido.

Después de definir la función de pertenencia lo siguiente es la definición de las implicaciones, que consiste en asignar una función de pertinencia a una agrupación antecedente-consecuente del tipo  $p \rightarrow q$  definida por la Ec.9, dichas relaciones de implicación son la base del razonamiento relacionados en reglas, en lógica difusa la metodología más utilizada es el propuesto por Mamdani (1977).

$$p \rightarrow q \equiv p^q \Rightarrow \mu_{p \rightarrow q}(u, v) = \min(\mu_A(u), \mu_B(v)) \quad \text{Ec. 1}$$

Dónde: u, v son universos de discurso.

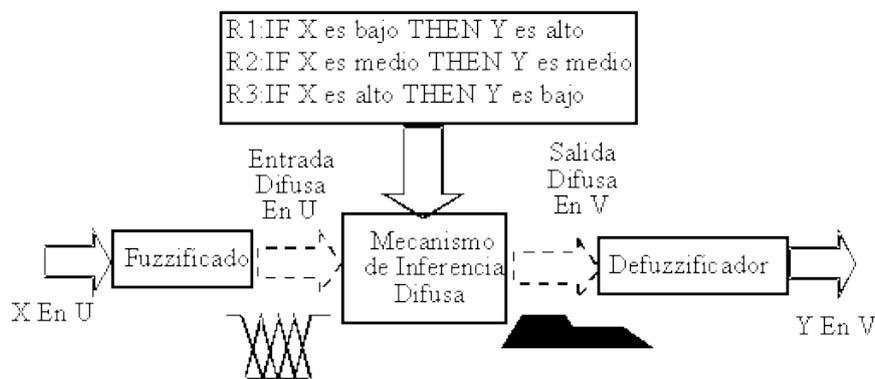
Por último se debe pasar por una transformación denominada defuzzificación que trata en pasar de una respuesta difusa a otra respuesta no difusa generada a partir del sistema para lograr esta transformación existen varios métodos y su elección depende del problema en cuestión. (Santana, 2013, p. 502).

### 1.1.2.3.2 Tipos de sistemas de lógica difusa

Los sistemas de lógica difusa tienen una estrecha relación con los conceptos difusos tales como conjuntos difusos, variables lingüísticas y demás. Los más populares sistemas de lógica difusa que se encuentran son de los siguientes tipos: sistemas difusos tipo Mamdani (con fuzzificador y defuzzificador) Sistemas difusos tipo Takagi-Sugeno

#### Sistemas tipo Mamdani

En la figura 1-1 se muestra la configuración básica de un sistema tipo Mamdani



**Figura 1-1:** Sistema difuso Mamdani proceso general  
Fuente: (Lopez, 2003)

En un sistema difuso tipo Mamdani se distinguen las siguientes partes:

**Fuzzificador:** La entrada de un sistema de lógica difusa tipo Mamdani normalmente es un valor numérico proveniente, por ejemplo, de un sensor; para que este valor pueda ser procesado por el sistema difuso se hace necesario convertirlo a un "lenguaje" que el mecanismos de interferencia pueda procesar. Esta es la función del fuzzificador, que toma los valores numéricos provenientes del exterior y los convierte en valores "difusos" que pueden ser procesados por el

mecanismo de inferencia. Estos valores difusos son los niveles de pertenencia de los valores de entrada a los diferentes conjuntos difusos en los cuales se ha dividido el universo de discurso de las diferentes variables de entrada al sistema.

**Mecanismo de inferencia difusa:** Teniendo los diferentes niveles de pertenencia arrojados por el fuzzificador, los mismos deben ser procesados para generar una salida difusa. La tarea del sistema de inferencia es tomar los niveles de pertenencia y apoyado en la base de reglas generar la salida del sistema difuso.

**Base de Reglas Difusas:** La base de reglas son la manera que tiene el sistema difuso de guardar el conocimiento lingüístico que le permiten resolver el problema para el cual ha sido diseñado. Estas reglas son del tipo IF-THEN.

Una regla de la base de reglas o base de conocimiento tiene dos partes, el antecedente y la conclusión como se observa en la figura 2-1



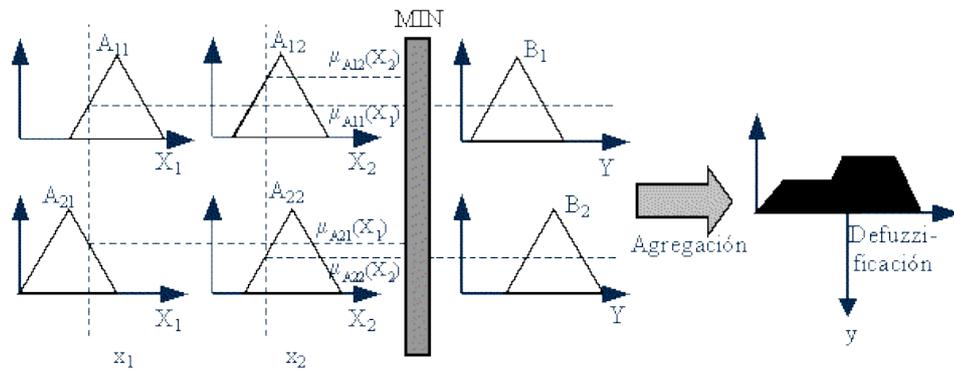
**Figura 2-1:** Regla de la base de conocimientos  
Fuente: (Lopez, 2003)

En un sistema difuso tipo Mamdani tanto el antecedente como el consecuente de las reglas están dados por expresiones lingüísticas.

**Defuzzificador:** La salida que genera el mecanismo de inferencia es una salida difusa, lo cual significa que no puede ser interpretada por un elemento externo (por ejemplo un controlador) que solo manipule información numérica. Para lograr que la salida del sistema difuso pueda ser interpretada por elementos que solo procesen información numérica, hay que convertir la salida difusa del mecanismo de inferencia; este proceso lo realiza el fuzzificador.

La salida del mecanismo de inferencia es un conjunto difuso resultante, para generar la salida numérica a partir de estos conjuntos existen varias opciones como el Centro de Gravedad, los Centros Promediados entre otros.

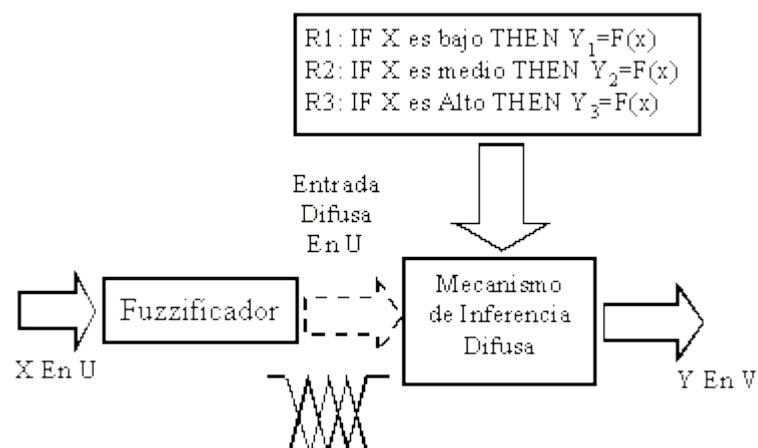
En la figura 3-1 se muestra de manera más detallada como es el funcionamiento de un sistema difuso Mamdani (Lopez, 2003)



**Figura 3-1:** Sistema Difuso Mamdani proceso detallado  
Fuente: (Lopez, 2003)

### Sistemas tipo Sugeno

Otra alternativa de procesamiento en los sistemas difusos fue la propuesta por Sugeno en los sistemas difusos que llevan su nombre. A continuación en la figura 4-1 se muestra el procesamiento general del sistema tipo Sugeno



**Figura 4-1:** Sistema Difuso Sugeno, procesamiento general  
Fuente: (Lopez, 2003)

En los sistemas difusos Sugeno se distinguen las siguientes partes:

**Fuzzificador:** Realiza la misma función que en los sistemas Mamdani explicados anteriormente

**Mecanismo de inferencia difusa:** Realiza la misma función que en los sistemas Mamdani explicados anteriormente

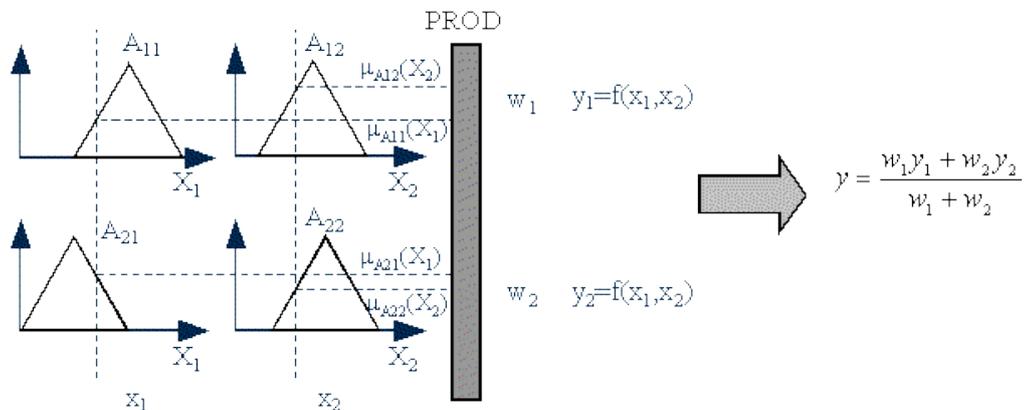
**Base de Reglas Difusas:** Las reglas de la base de conocimiento de un sistema Sugeno es diferente a las de los sistemas Mamdani pues el consecuente de estas reglas ya no es una etiqueta lingüística sino que es una función de la entrada que tenga el sistema en un momento dado, esto se ilustra en la figura 5-1.



**Figura 5-1:** Regla de base de conocimiento de sistema Sugeno  
Fuente: (Lopez, 2003)

En los sistemas difusos tipo Sugeno, los valores que arrojan los consecuentes de las diferentes reglas que se han activado en un momento determinado ya son valores numéricos por lo que no se necesita una etapa de defuzificación.

En la figura 25-1 se muestra de manera más detallada como es el funcionamiento de un sistema difuso Sugeno. (Lopez, 2003)



**Figura 6-1:** Regla de base de conocimiento de sistema Sugeno  
Fuente: (Lopez, 2003)

La técnica de inteligencia artificial que se ocupa para la elaboración del proyecto es lógica difusa, dentro de lo cual se decide trabajar con el sistema de tipo Mamdani debido su forma de expresar el resultado, los sistemas tipo Mamdani poseen una salida de tipo lógica a diferencia del sistema tipo Sugeno el cual nos da como salida una función matemáticas de las variables de ingreso.

## **1.2 Visión Artificial**

### **1.2.1 *Concepto***

La visión artificial es una disciplina científica que incluye métodos para adquirir, procesar y analizar imágenes del mundo real con el fin de producir información que pueda ser tratada por una máquina.

Una manera simple de comprender este sistema es basarnos en nuestros propios sentidos. Los humanos usamos nuestros ojos para comprender el mundo que nos rodea, y la visión artificial trata de producir ese mismo efecto en máquinas. Éstas podrán percibir y entender una imagen o secuencia de imágenes y actuar según convenga en una determinada situación. La comprensión en los dispositivos se consigue gracias a una descomposición de la imagen en pequeños fragmentos (píxeles) y en su posterior estudio.

La principal finalidad de la visión artificial es dotar a la máquina la capacidad para ver lo que ocurre en el mundo real, y así poder tomar decisiones para automatizar cualquier proceso. (contaval, 2016)

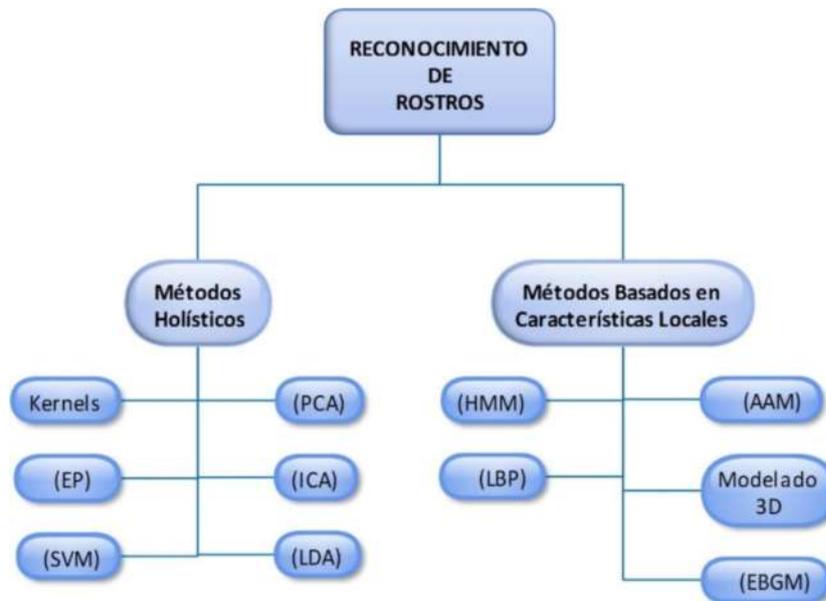
Existen muchas tecnologías que utilizan visión artificial: como la detección de objetos, el reconocimiento de objetos, la restauración de imágenes y la reconstrucción de una escena, etc.

### **1.2.2 *Reconocimiento Facial***

El reconocimiento facial es una aplicación dirigida por ordenador que identifica automáticamente a una persona en una imagen digital. Esto es posible mediante un análisis de las características faciales del sujeto extraídas de la imagen o de un fotograma clave de una fuente de video, y comparándolas con una base de datos.

### 1.2.2.1 Algoritmos de visión artificial aplicados al reconocimiento facial.

Los algoritmos de reconocimiento facial se clasifican en dos grupos: métodos holísticos y métodos basados en características locales. En la figura 7-1 se puede ver los diferentes métodos de reconocimiento que conforma cada grupo.



**Figura 7-1:** Algoritmos de reconocimiento facial  
Realizado por: JARA, Caroline; CABA, Carlos, 2018

#### 1.2.2.1.1 Métodos holísticos

Utilizan toda la imagen de la cara como entrada al sistema de reconocimiento, siendo ésta la unidad básica de procesamiento.

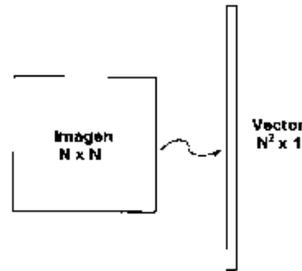
#### **PCA (Análisis de Componentes Principales)**

Este método se basa en un sub espacio de imágenes también conocido como eigenfaces, Una imagen de  $n \times n$  píxeles es representada como una matriz de dimensiones  $N \times N$  píxeles, donde cada píxel posee un valor de intensidad diferente (valores entre 0 y 255) que se obtiene al transformar la imagen del tipo RGB (rojo, verde y azul) a escala de grises. Las imágenes están formadas por coordenadas:  $I(x,y)$ .

La idea principal del método es encontrar los vectores que mejor representen la distribución de las imágenes, los cuales definen un sub-espacio correspondiente a las imágenes de rostros de seres humanos. Ese sub-espacio es llamado espacio de rostros. El proceso comienza con una

imagen de rostro  $I(x,y)$  que se puede representar en dos dimensiones  $N \times N$  (matriz de valores de intensidad de 8-bits).

Una imagen puede ser considerada como un vector de dimensiones  $N^2$ . Una típica imagen de  $256 \times 256$  píxeles puede ser un vector de una dimensión de  $65536$ . En la figura 8-1 se puede ver el proceso de transformación de una imagen de  $N \times N$  dimensiones a un vector  $N^2 \times 1$ .



**Figura 8-1:** Imagen  $N \times N$  transformada a vector  $N^2 \times 1$   
Fuente: (Villalón, 2012, p. 18)

El objetivo del análisis de componentes principales es encontrar los vectores que mejor almacenen la distribución de las imágenes de rostros en el espacio completo de imágenes.

#### **Pasos del algoritmo de reconocimiento de rostro utilizando PCA basado en Eigenfaces:**

Almacenar un conjunto de imágenes de entrenamiento de diferentes personas, pudiéndose tener subconjuntos de imágenes para cada persona que contengan diferentes posturas, condiciones de iluminación, etc.

Crear una matriz formada por la imagen de entrada y las almacenadas en la base de datos. Mediante el proceso arriba descrito, se calculan los eigenvectores mediante la matriz de covarianza.

Una vez obtenidos los vectores característicos se comparan las distancias entre el vector que representa a la imagen original con el resto.

Establecido un umbral de reconocimiento si el menor valor del paso anterior es menor que el umbral, la imagen de la cara de entrada es considerada como conocida, si es mayor, se considera desconocida. De aquí se desprende que los valores de las distancias dependen en cierta medida del tamaño de la base de datos, puesto que la matriz de covarianza y los vectores característicos son calculados a partir de la matriz formada por la imagen de entrada y las ya almacenadas, por lo que el umbral ha de ser dinámico y adaptarse según la variación de las distancias entre caras,

o lo que es lo mismo, según el tamaño de la base de datos que almacena las imágenes de las caras. (Villalón, 2012, p. 21).

### **LDA (Análisis Lineal Discriminante)**

LDA es conocida como una aproximación estadística que consiste en clasificar muestras de clases desconocidas basadas en ejemplos de entrenamiento con clases conocidas, esta técnica maximiza la varianza entre clases y minimizar la varianza entre cada clase. (Pereyra, p. 44).

En la figura 9-1 se puede observar que hay grandes variaciones entre clases y a la vez pequeñas en cada clase.



**Figura 9-1:** Ejemplo de seis clases usando LDA  
Fuente: (Blanco; Chávez, 2012, p. 42)

Esta técnica consiste en discriminar de forma lineal las características de los rostros. Para lo cual obtiene las características con mayor información en las caras para su luego clasificarlas. Este proceso se hace mediante la extracción de características basadas en proyección, al encontrar el espacio subyacente que mejor discrimina entre las clases y posteriormente se clasifica. El objetivo es minimizar la separación dentro de una misma clase a la vez que se maximiza la separación entre clases diferentes. (Pérez, 2009, p. 31).

### **ICA (Análisis de Componentes Independientes)**

El algoritmo ICA toma en cuenta los estadísticos de orden superior con lo cual se obtiene una representación de los datos más independiente y por tanto, potente. Halla un conjunto de bases de forma que minimiza las dependencias de órdenes superiores para conseguir que la información proyectada sobre dicha base sea estadísticamente independiente. (Pérez, 2009, p. 32).

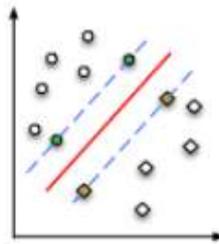
### **SVM (Máquinas de vectores de Soporte)**

Las máquinas de vectores soporte (SVM) fueron introducidas por Vapnik (1990) y sus colaboradores Boser (1992), Cortes & Vapnik (1995). Originariamente las SVMs fueron

pensadas para resolver problemas de clasificación binaria, actualmente se utilizan para resolver otros tipos de problemas como regresión, agrupamiento, multclasificación.

También han sido utilizadas con éxito en diversos campos, tales como visión artificial, reconocimiento de caracteres, categorización de texto e hipertexto, clasificación de proteínas, procesamiento de lenguaje natural, análisis de series temporales.

SVM permite determinar si un vector de entrada pertenece o no a un grupo definido disponiendo de dos posibles grupos y cada vector de entrada pertenece a un grupo, para lo cual busca un plano que separe los puntos pertenecientes a cada categoría. En la figura 10-1 se puede ver el plano de separación que es representado por una línea continua y los puntos más cercanos al plano de separación pertenecen a la línea discontinua y forman el llamado vector soporte. (González, 2015, p. 12)



**Figura 10-1:** Separación de puntos línea discontinua  
Fuente: (González, 2015, p. 12)

### 1.2.3 Métodos basados en características locales

Se extraen características locales, como ojos, nariz, boca, etc. Sus posiciones y estadísticas locales constituyen la entrada al sistema de reconocimiento.

#### AAM (Modelo de Apariencia Activa)

Se basa en la adaptación de un modelo estadístico creado durante una fase de entrenamiento a partir de una serie de puntos de referencia. Para cada imagen se crea una malla de puntos característicos y adopta la forma y apariencia del objeto de la región de interés, después para todas se realiza un análisis PCA para obtener las variaciones y poder obtener la malla modelo que luego se aplicara para la extracción como se puede ver la figura 11-1. (Zapatero, 2016, p. 9).



**Figura 11-1:** Método AAM

**Fuente:** (Zapatero, 2016, p. 9)

### **Modelado 3D**

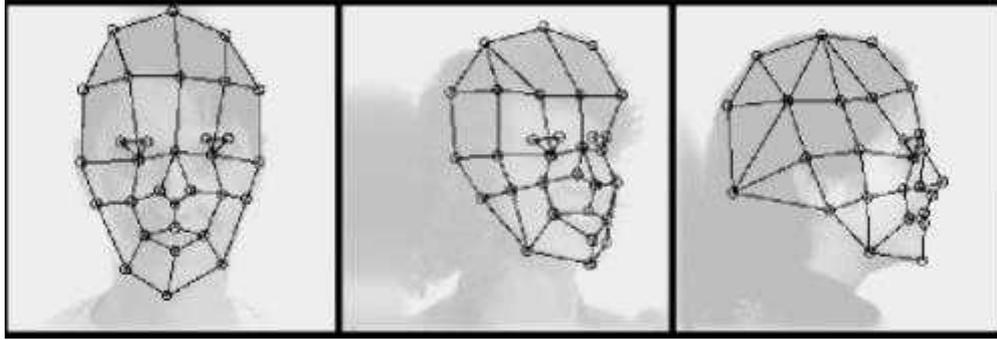
Esta técnica utiliza una malla en 3D para identificar los puntos clave hallando los puntos de máxima curvatura en los perfiles. Dichos puntos y su relación entre ellos son luego representados en un grafo.

### **EBGM (Elastic Bunch Graph Matching)**

Este algoritmo usa grafos en estrella apoyado en la técnica de grafos en 2D, esto debido a que los rostros comparten una estructura topológica similar, cuyos nodos son puntos clave (ojos, narices,..) y las aristas las distancias entre ellos. (Pérez, 2009, p.33).

Este trabajo se desarrollara con este algoritmo EBGM porque una de las características de este algoritmo es que trabaja con algunos puntos de interés del rostro y no con la cara en su totalidad, también no necesariamente se necesita una captura de imagen frontal de la persona que se va a reconocer sino que puede tener como entrada varias imágenes en distintas posiciones, en nuestro trabajo se tomará como imagen de entrada una en posición frontal y otra de perfil.

La técnica de EBGM se desarrolla básicamente en dos etapas: la primera consiste en ajustar un grafo de puntos principales a la cara del individuo como se puede ver en la figura 12-1, utilizando para ello un modelo estadístico de dicho grafo; la segunda etapa extrae características locales en dichos puntos y halla la distancia entre el grafo obtenido y sus descriptores al grafo almacenado de la persona a identificar. Dependiendo de la distancia encontrada, se ratifica o no la identidad del individuo.



**Figura 12-1:** Método EBGM

Fuente: (Zapatero, 2016, p. 9)

La implementación del algoritmo EBGM requiere el uso de las wavelets Gabor (filtros espaciales paso-banda, que permiten alcanzar la resolución conjunta de información máxima en los espacios bidimensionales espacial y frecuencial), ya que estas brindan una descripción de la información en frecuencia, en una región específica de la imagen. Para ello es necesario realizar la operación de convolución de la región con una variedad de wavelets; estas wavelets son almacenadas como máscaras en arreglos bidimensionales. Las máscaras se pueden definir mediante la ecuación 1 y 2, donde las wavelets se separan en parte par y parte impar.

$$PAR. Ond_p(x, y) = e^{-\frac{(x-x_0)^2+y^2*(y-y_0)^2}{2\sigma^2}} * \cos\left(\frac{2*\pi*x_r}{\lambda}\right) \quad Ec. 1$$

$$IMPAR. Ond_i(x, y) = e^{-\frac{(x-x_0)^2+g^2*(y-y_0)^2}{2s^2}} * \sin\left(\frac{2*\pi*x_r}{\lambda}\right) \quad Ec. 2$$

Con:  $x_r = (x - x_0)^2 * \cos(\sigma) + (y - y_0)^2 * \sin(\sigma)$  donde define la longitud de onda,  $\sigma$  define el radio de la gaussiana,  $Y$  define la relación de aspecto de la gaussiana,  $\theta$  representa la orientación de la portadora y  $(x_0, y_0)$  determinan la ubicación del valor pico de la función gaussiana.

A continuación se describe la parte del algoritmo EBGM que se refiere a la localización de puntos característicos.

Se define una estructura de grafo sobre la cara, cuyos nodos son puntos de interés, que se puedan localizar fácilmente y que posean la misma estructura en todos los rostros.

Cada nodo del grafo es caracterizado utilizando un banco de filtros de Gabor de diferentes frecuencias y orientaciones. En cada nodo, se calcula la respuesta de todos los filtros, denominado Jet. Por consiguiente, cada nodo queda etiquetado con sus coordenadas y su Jet asociado.

Para una imagen nueva se busca en la base de datos el conjunto de Jets que sean más similares. Para realizar esto, se centra el grafo en los ojos de la nueva imagen, se calculan los jets de estos puntos y se comparan con los jets de los ojos de la base de datos. Para la comparación de los jets se utiliza una función de similitud de fase, ésta es similar a una correlación, y está definida como se describe en la ecuación 3.

$$S(J, J') = \frac{\sum_{j=1}^n a_j a'_j \cos(\phi_j - \phi'_j)}{\sqrt{\sum_{j=1}^n a_j^2 \sum_{j=1}^n a'^2_j}} \quad \text{Ec. 3}$$

Dónde:  $a'$  es la magnitud del jet de la base de datos,  $a$  la magnitud del Jet de la imagen nueva,  $\phi_j$  la fase del Jet de la base de datos,  $\phi'_j$  la fase del Jet de la imagen nueva y  $n$  el número de coeficientes del Jet.

Este algoritmo tiene en cuenta que las imágenes faciales reales tienen muchas características no lineales que no son tratadas en los métodos lineales de análisis (LDA, PCA), tales como variaciones en la iluminación (iluminación de exteriores vs interior fluorescente), postura (frontal vs inclinada) y expresión (sonrisa vs ceño fruncido). Sin embargo, la dificultad de este método es el requerimiento de la precisa localización del punto de referencia. (Pérez, 2013, p. 7-8).

### **HMM (Modelo Oculto de Markov)**

Un HMM puede ser considerado como la red bayesiana más simple, están basados en la propiedad de Markov por el cual la distribución de probabilidad de un proceso del valor futuro de una variable únicamente depende de su valor presente, independientemente de los valores pasados de la misma. Se trata de un modelo en el cual el estado de cada nodo es desconocido, es decir, está oculto. (Zapatero, 2016, p. 17)

### **LBP (Patrones Binarios Locales)**

Es utilizado para detectar todo tipo de objetos basados en texturas, invariante ante cambios de iluminación y capaz de detectar movimiento.

Este sistema es capaz de extraer la estructura local de una imagen a partir del entorno de cada uno de los píxeles que lo conforman, el proceso consiste en ir comparando el valor de cada píxel con el de sus vecinos distribuidos en una circunferencia a su alrededor. (Zapatero, 2016, p. 6-7)

### 1.2.3.1 Estructura de un sistema de reconocimiento facial

En toda aplicación de reconocimiento facial, se debe tener en cuenta sus diferentes etapas, el sistema de reconocimiento recibirá como entrada imágenes de la persona en dos posiciones frontal y perfil para luego identificar el rostro en cada imagen de entrada para ser procesadas por nuestro sistema para después hacer el reconocimiento, las características tomadas de las imágenes de entrada se compara con las imágenes de la base de datos para verificar la identidad de la persona. La figura 13-1 muestra un diagrama de las etapas que conforman el sistema las cuales serán explicadas a continuación:



**Figura 13-1:** Diagrama general del reconocimiento facial  
Realizado por: JARA, Caroline; CABA, Carlos, 2018

#### 1.2.3.1.1 Imagen de entrada

Para las imagen de entrada tanto de frente como de perfil tenemos que tener en cuenta una sola consideración, nos referimos a la correcta posición, es decir que la persona debe estar ubicada lo mejor orientada posible hacia la cámara.

#### 1.2.3.1.2 Detección del rostro

Detecta la cara de una persona en imágenes o videos de manera rápida y eficiente, el cual consiste en reconocer en qué zona de la imagen se encuentra una cara, dando lugar a la segmentación (extracción de dicho rostro de la imagen).

Para poder encontrar el rostro en una imagen existen métodos de detección facial que detallamos a continuación.

### **Métodos de detección facial**

**Basados en ambiente controlado:** Como su nombre lo indica las imágenes que maneja este método son tomadas en ambientes controlados, y debido a que se tiene un control en la imagen para la detección se aplican sencillos procesos de detección de bordes.

**Basados en el color de piel:** En las imágenes los colores vienen representados por vectores de iluminancia y crominancia, para esta detección lo que se realiza es una búsqueda de los componentes de color que más se acerquen a los datos de colores de piel conocidos. Este método es dependiente del control de luminancia y fondo de la imagen, además posee muchas dificultades en distinguir todos los rostros debido a que el rango de colores de piel es muy amplio.

**Basados en movimiento:** Este método es único ya que como método de entrada posee capturas de video y no solo imágenes, su funcionamiento está basado en que por una cámara de video fija muy pocas cosas van a captarse y de entre ellas la mayoría serán rostros. También existen variantes en las cuales las tomas se realizan con cámaras de video móviles pero sus algoritmos de funcionamiento son más complejos. Estos métodos pueden combinarse con algoritmos adicionales para poseer una mejor detección.

**Basados en plantillas:** Utiliza un modelo de rostro o plantilla y trabaja buscando en la imagen de entrada cualquier posible coincidencia con la plantilla.

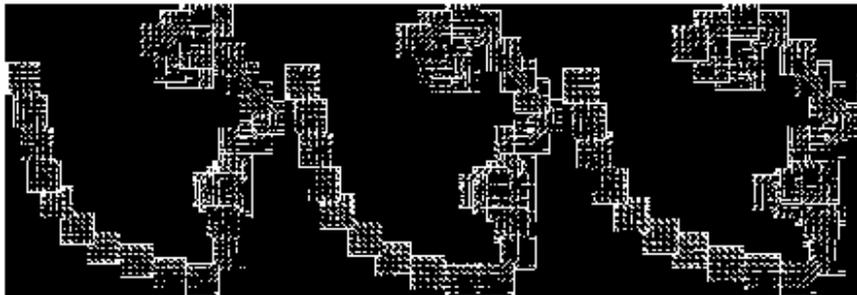
**Basados en el aspecto:** Por medio de etapas de entrenamiento logran obtener las variaciones del aspecto del individuo y proceden a realizar el reconocimiento.

**Basados en boosting:** Al hablar de boosting se refiere a obtener un clasificador u seleccionador completo creado a partir de clasificadores sencillos, un ejemplo que resalta en el mercado es el Método de Viola-Jones el cual utiliza boosting adaptivo

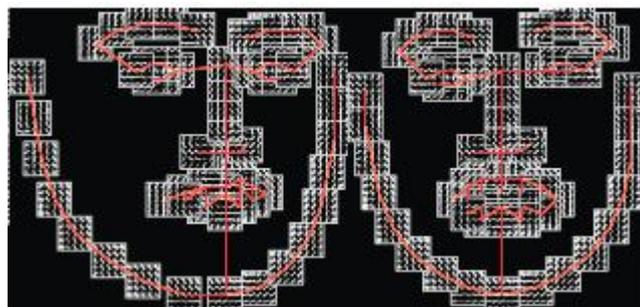
**Otros métodos utilizan la similitud del contorno de un rostro con una elipse:** Se trata de adaptar una porción de la imagen de entrada a una pequeña figura en forma de elipse y mostrar los resultados obtenidos, debido a esta manera de procesar es el método más susceptible a iluminación, postura, tamaño y calidad de imágenes de entrada

### **Face Detection, Pose Estimation, and Landmark Localization in the Wild**

En este proyecto se utilizó un modelo unificado para detección de rostros, estimación de postura, y la estimación de puntos de referencia en el mundo real, propuesto por Xiangxin Zhu (2012) y Deva Ramanan (2012) del Departamento de Ciencias de la Computación de la Universidad de California, Irvine, presentando un modelo unificado para la detección de rostros, planteando la estimación, y la estimación de punto de referencia en el mundo real. El modelo se basa en una mezcla de árboles con un conjunto compartido de partes; modelando cada hito facial como una parte y el uso de mezclas globales para captar los cambios topológicos debido al punto de vista. Los modelos con estructura de árbol son sorprendentemente eficaces en la captura de deformación elástica global siendo fácil de optimizar. (Zhu; Ramanan, 2012, p. 2).



**Figura 14-1:** Modelo unificado para la detección del rostro  
Fuente: (Zhu; Ramanan, 2012, p. 2).



**Figura 15-1:** Modelo unificado para la detección del rostro  
Fuente: (Zhu; Ramanan, 2012, p. 2).

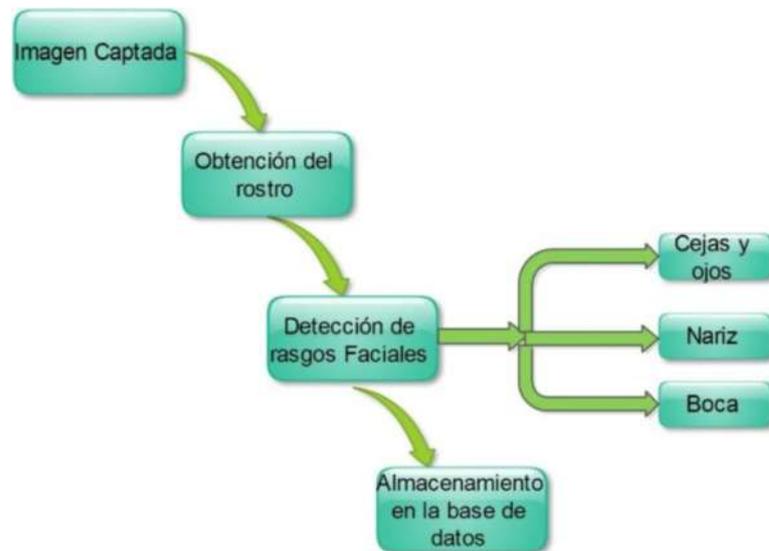
El Unified Model For Face Detection, Pose Estimation, and Landmark Estimation in Real-World, Xiangxin Zhu (2012) y Deva Ramanan (2012), codifica los cambios topológicos debido al punto de vista. Las líneas rojas que se muestra en las figura 14-1 y 15-1 indican resortes entre pares de partes; teniendo en cuenta que no hay bucles cerrados, manteniendo la propiedad del árbol. Todos los árboles hacen uso de un conjunto común compartido de plantillas de partes, lo que hace que el aprendizaje y la inferencia sean eficientes.

Según Xiangxin Zhu (2012) y Deva Ramanan (2012), en términos de detección de rostros, el modelo supera sustancialmente a Viola-Jones y está a la par de los sistemas comerciales. En términos de estimación de postura y punto de referencia, los resultados dominan incluso los sistemas comerciales. Los resultados son particularmente impresionantes ya que el modelo está entrenado con cientos de caras, mientras que los sistemas comerciales usan hasta miles de millones de ejemplos. Otro resultado es la evidencia de una gran brecha entre las soluciones académicas actualmente disponibles y los sistemas comerciales; abordando este problema lanzando un software de código abierto. (Zhu; Ramanan, 2012, p. 2).

#### 1.2.3.1.3 *Normalización*

El siguiente paso después de detectar el rostro en la imagen es la normalización de la cara detectada. Esto significa que todas las imágenes que tenemos deben ser de igual tamaño y de aproximadamente la misma calidad, con respecto a las imágenes que se tiene en la base de datos.

Dentro de la normalización se realizó la detección de rasgos faciales de la cara las mismas que también se encuentran en la base de datos. En la figura 16-1 se puede ver un diagrama de cuales fueron los rasgos faciales detectados en el rostro.



**Figura 16-1:** Detección de rasgos faciales  
 Realizado por: JARA, Caroline; CABA, Carlos, 2018

El reconocimiento puede ser factible únicamente cuando todas las imágenes, sean idénticas en tamaño y calidad relativos a las imágenes de la base de datos. La normalización incrementa las posibilidades de reconocimiento. (Introna; Nissenbaum, 2010, p.16).

#### 1.2.3.1.4 *Extracción de características*

Luego de haber realizado el reconocimiento el siguiente punto es elaborar la denominada plantilla biométrica que se lo realiza extrayendo los principales puntos característicos del rostro. Esta denominada plantilla es la base de cualquier método de reconocimiento.

Para esto se utilizan Filtros Bidimensionales de Gabor (FBG) aplicados en cada porción característica de las imágenes, esto se lo realiza para obtener vectores característicos de cada porción la cual será única y exclusiva para cada persona.

Los filtros de Gabor están conformados por grupo de wavelets, es decir cada conjunto de frecuencia y orientación se lo denomina un wavelets. Estos filtros son comúnmente utilizados para los métodos de reconocimiento y detección de bordes. Y abarcan casi cualquier caso de reconocimiento ya que poseen la facilidad de varias su orientación. (Guimarães, 2017, p.20-21).

Las funciones Bidimensionales de Gabor están conformados por cuatro parámetros, de los cuales dos expresan su localización en el dominio espacial (x,y), y otros dos que expresan la frecuencia espacial de sintonía (F) y la orientación ( $\Phi$ ) o en coordenadas cartesianas, frecuencia espacial en dirección horizontal (u) y dirección vertical (v), las cuales indican la localización en el dominio de la frecuencia.

La señal del FBG en coordenadas espaciales está localizada en  $(x=0,y=0)$  y en el dominio de la frecuencia se encuentra localizada a las frecuencias (F) y orientaciones (f) a las que se encuentre sintonizada, estas coordenadas reciben el nombre de coordenadas polares.

Las funciones del FBG pueden expresarse analíticamente mediante la Ec.4:

$$h(x, y) = g(x', y') * \exp(2pjFx') \quad \text{Ec. 4}$$

Las funciones de Gabor operan en el conjunto de los números complejos, y que la parte real es la función de Gabor simétrica (simetría par o fase coseno). En el conjunto de los números reales, el FBG puede calcularse asignando a la fase de modulación de la luminancia el valor  $\Phi = 0^\circ$ , con lo que resulta una modulación senoidal. La parte imaginaria es la función de Gabor asimétrica (simetría impar o fase seno), la cual se puede calcular en el conjunto de los reales asignando a la fase de modulación de la luminancia el valor  $\Phi = 90^\circ$ , con lo que resulta una modulación cosenoidal. La parte real (simetría par) e imaginaria (simetría impar), se encuentran definidas por las ecuaciones Ec.5, Ec.6, Ec.7 y Ec.8:

$$(x', y') = (x \cos f + y \sin f, -x \sin f + y \cos f) \quad \text{Ec. 5}$$

$$h(x, y) = h_c(x, y) - j h_s(x, y) \quad \text{Ec. 6}$$

$$h_c(x, y) = g(x', y') \cos(2pFx') \quad \text{Ec. 7}$$

$$h_s(x, y) = g(x', y') \sin(2pFx') \quad \text{Ec. 8}$$

Donde  $(x', y')$  es la rotación del ángulo ( $\Phi$ ) en la parte real ( $x'$ ) e imaginaria ( $y'$ ) de la señal elemental de Gabor,  $h_c(x,y)$  es la señal elemental de Gabor con componentes reales (simetría par) y  $h_s(x,y)$  es la señal elemental de Gabor con componentes imaginarios (simetría impar). (Oropeza, 2008, p.68).

#### 1.2.3.1.5 Reconocimiento (Comparación).

Para la comparación se ocupa la distancia euclídea que es la disimilaridad más conocida y más sencilla de comprender, pues su definición coincide con el concepto más común de distancia.

La distancia euclídea, a pesar de su sencillez de cálculo y de que verifica algunas propiedades interesantes tiene dos graves inconvenientes:

El primero de ellos es que la euclídea es una distancia sensible a las unidades de medida de las variables: las diferencias entre los valores de variables medidas con valores altos contribuirán en mucha mayor medida que las diferencias entre los valores de las variables con valores bajos. Como consecuencia de ello, los cambios de escala determinarán, también, cambios en la distancia entre los individuos. Una posible vía de solución de este problema es la tipificación previa de las variables, o la utilización de la distancia euclídea normalizada.

El segundo inconveniente no se deriva directamente de la utilización de este tipo de distancia, sino de la naturaleza de las variables. Si las variables utilizadas están correlacionadas, estas variables nos darán una información, en gran medida redundante. Parte de las diferencias entre los valores individuales de algunas variables podrían explicarse por las diferencias en otras variables. Como consecuencia de ello la distancia euclídea inflará la disimilaridad o divergencia entre los individuos.

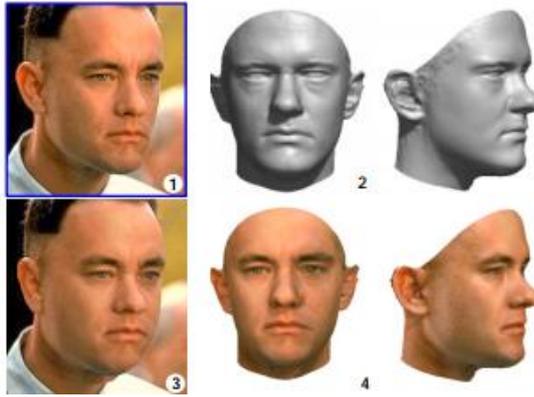
La distancia euclídea será, en consecuencia, recomendable cuando las variables sean homogéneas y estén medidas en unidades similares y/o cuando se desconozca la matriz de varianzas. (valencia, 2016)

#### 1.2.3.2 *Resultados obtenidos*

El resultado de cualquier sistema de reconocimiento siempre será la identidad de la persona a la cual se analiza, solo que suele manifestarse de diferentes formas como el porcentaje de similitud, la identidad de la persona, el resultado de búsqueda con personas de una base de datos, etc.

#### **1.2.4 *Modelo Facial en 3D***

Los modelos 3D Morphable (3DMM) son poderosos modelos estadísticos 3D de la forma y la textura de la cara humana. Originalmente Blanz y Vetter (1999), presentaron su trabajo en el cual se demostró que un 3DMM utilizado en un marco de análisis por síntesis es capaz de inferir una superficie facial en 3D completa a partir de una sola imagen de una persona como se puede observar en la figura 17-1. Desde entonces, los 3DMM se han aplicado ampliamente en numerosas áreas, como la visión artificial, el análisis del comportamiento humano, los gráficos por computadora y la cirugía craneofacial. (Booth; Roussos; Zafeiriou; Ponniah, Dunaway, 2016, p.1).



**Figura 17-1:** Coincidencia de un modelo morphable  
**Fuente:** (Blanz Vetter, 1999, p. 192)

Un 3DMM se construye mediante la realización de alguna forma de reducción de dimensionalidad, típicamente Análisis de Componentes Principales (PCA), en un conjunto de entrenamiento de mallas faciales. Esto es factible si y solo si cada malla se reparametriza primero en una forma consistente donde el número de vértices, la triangulación y el significado anatómico de cada vértice se hacen consistentes en todas las mallas. Se dice que las mallas que satisfacen las propiedades anteriores están en una densa correspondencia entre ellas. Si bien este problema de correspondencia es fácil de enunciar, es difícil resolver de forma precisa y robusta entre las mallas faciales altamente variables.

#### 1.2.4.1 *Construcción 3DMM*

La construcción de un 3DMM ocurre en dos etapas principales: correspondencia densa y alineación de similitud y modelado estadístico.

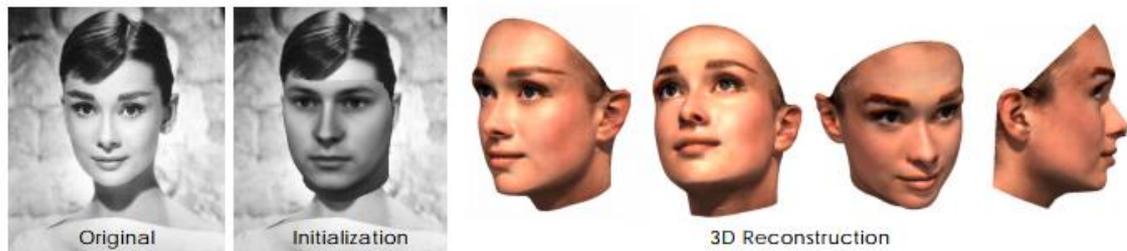
#### 1.2.4.2 *Correspondencia densa*

Una colección de mallas se reparametriza en una forma en la que cada malla tiene el mismo número de vértices unidos en una triangulación que se comparte entre todas las mallas. Además, el significado semántico o anatómico de cada vértice se comparte en toda la colección.

#### 1.2.4.3 *Alineación de similitud y modelado estadístico*

La recolección de mallas en correspondencia densa se somete a un Análisis Procrustes para eliminar los efectos de similitud, dejando solo información de forma. Las mallas procesadas se analizan estadísticamente, generalmente con PCA, generando un modelo 3D deformable como una base lineal de formas.

En la figura 18-1 podemos ver una representación de la creación del modelo 3D.



**Figura 18-1:** Combinación automática a un modelo morfálico

Fuente: (Banz Vetter, 1999, p. 193)

### 1.3 Hardware

#### 1.3.1 Controlador (Computador portátil TOSHIBA Satellite S855)



**Figura 19-1:** Portatil TOSHIBA Satellite S855.

Fuente: (Toshiba America Information System, 2018) Toshiba

Uno de los dispositivos más importantes del proyecto es sin duda el controlador, es decir el dispositivo donde se ejecuta todos los procesos, y para ello se considera utilizar un computador similar al de la figura 19-1 debido a que este tipo de trabajo es fijo y bajo la necesidad de procesamiento más rápido.

A pesar de que en el mercado los controladores que existen son de capacidades iguales o en algunos casos superiores a un computador se considera un desperdicio invertir en un controlador que ha sido diseñado para miniaturización y para trabajos portátiles teniendo a la mano un computador que nos abastece de forma eficiente para los fines pertinentes.

También una razón importante y una ventaja de usar el computador es la facilidad de ejecutar la interface de usuario y permitir su manipulación de una forma cómoda, debido a que en la parte del registro al sistema se registra con datos personales e interactúa con el sistema. A su vez también ayuda para la opción de vigilancia en tiempo real que posee el proyecto.

En la tabla 1-1 se puede observar las especificaciones del computador portátil TOSHIBA Satellite S855.

**Tabla 1-1:** Especificaciones de Computador Portátil TOSHIBA Satellite S855

<b>Marca</b>	<b>Toshiba</b>
<b>Modelo</b>	Satellite S855-S5378
<b>Procesador</b>	Intel Core i7-3630QM 2.4GHz (3.4GHz c/TB)
<b>Memoria RAM</b>	8 GB DDR3
<b>Disco Duro</b>	750 GB
<b>Pantalla</b>	LED 15.6" HD 1366 x768
<b>Unidad óptica</b>	DVD ± RW
<b>Tarjeta de video</b>	Intel
<b>Conectividad</b>	Gigabit Ethernet (RJ-45) Wi-Fi 802.11b/g/n Bluetooth-4.0
<b>Multimedia</b>	Cámara web integrada Entrada para auriculares estéreo/micrófono Altavoces
<b>Puertos y Ranuras</b>	VGA HDMI USB 2.0 (Sleep and Charge) (x1) USB 3.0 Network (RJ-45) Lector de tarjetas de memoria Audio in/out
<b>S.O.</b>	Windows 7
<b>Batería</b>	6 celdas - ion de litio
<b>Peso</b>	2.49 Kg

Realizado por: JARA, Caroline; CABA, Carlos 2018

### 1.3.2 Web Cam Logitech C270



**Figura 20-1:** Cámara Web Logitech C270.

Fuente: (Logitech, 2018) Logitech.

En el mercado actualmente existen un número limitado de webcams como dispositivos individuales, el futuro de éstas es desaparecer ya que actualmente son muy bien reemplazadas por las Webcam integradas en los computadores portátiles o también por los Smartphone que

actualmente vienen integrados de tanta tecnología que es raro ver una Webcam como accesorio independiente.

Con estas razones se decide no ocupar dichas cámaras, pero dadas las circunstancias en las que se desarrolla el sistema se ocupa la Webcam Logitech C270 como se puede observar en la figura 20-1, esta cámara se ocupa por las siguientes razones:

La razón por la que se ha elegido esa marca en especial es la resolución que brinda y que se usa en el proyecto (resolución de 800 x 600).

Otra razón es que la cámara a diferencia de sus marcas competidoras es más conveniente en cuanto a financieramente se refiere, y también por su disponibilidad en el mercado.

Por último el sistema de reconocimiento necesita un tiempo de respuesta corto y exige una transmisión de información sin ningún tipo de pérdida.

En la tabla 2-1 se puede ver las especificaciones de las cámara web Logitech C270.

**Tabla 2-1:** Especificaciones de cámara web Logitech C270

<b>Videoconferencias</b>	HD (1280 x 720 píxeles) con el sistema recomendado
<b>Captura de Video</b>	Hasta 1280 x 720 píxeles
<b>Tecnología</b>	Logitech Fluid Crystal™
<b>Fotos</b>	Hasta 3.0 megapíxeles (mejora por software)
<b>Certificación</b>	USB 2.0 de alta velocidad
<b>Clip universal</b>	Monitores, LCD, CRT o portátiles

Fuente: (Logitech, 2018) Logitech.

### 1.3.3 Arduino Mega 2560



**Figura 21-1:** Logo Arduino Mega 2560.

Fuente: (Thayer, 2018).

La placa Arduino Mega se utiliza en el proyecto con la finalidad de establecer una conexión física y lógica entre el procesador y los distintos dispositivos de entrada y salida del proyecto como la cerradura eléctrica, el conjunto de elementos de acceso (botón, led indicador, placa).

Arduino Mega es una tarjeta de desarrollo open-source construida con un microcontrolador modelo Atmega2560 que posee pines de entradas y salidas (E/S), analógicas y digitales. Esta tarjeta es programada en un entorno de desarrollo que implementa el lenguaje Processing/Wiring. Arduino puede utilizarse en el desarrollo de objetos interactivos autónomos o puede comunicarse a un PC a través del puerto serial (conversión con USB) utilizando lenguajes como Flash, Processing, MaxMSP, etc.

El Arduino Mega tiene 54 pines de entradas/salidas digitales (14 de las cuales pueden ser utilizadas como salidas PWM), 16 entradas analógicas, 4 UARTs (puertos serial por hardware), cristal oscilador de 16MHz, conexión USB, jack de alimentación, conector ICSP y botón de reset. Arduino Mega incorpora todo lo necesario para que el microcontrolador trabaje; simplemente conéctalo a tu PC por medio de un cable USB o con una fuente de alimentación externa (9 hasta 12VDC). El Arduino Mega es compatible con la mayoría de los shields diseñados para Arduino Duemilanove, diecimila o UNO. (Thayer, 2018).

En la tabla 3-1 se puede ver las especificaciones de Arduino Mega.

**Tabla 3-1:** Especificaciones de arduino Mega 2560

<b>Microcontrolador</b>	<b>ATmega2560</b>
<b>Voltaje operativo</b>	5 V
<b>Voltaje de entrada</b>	7-12 V
<b>Voltaje de entrada (límite)</b>	6-20 V
<b>Pines E/S digitales</b>	54 (15 con salida PWM)
<b>Pines entrada analógica</b>	16
<b>Intensidad por pin E/S</b>	40 mA
<b>Intensidad en pin 3.3V</b>	50 mA
<b>Memoria flash</b>	256 KB (8 KB para gestor de arranque "bootloader")
<b>SRAM</b>	8 KB
<b>EEPROM</b>	4 KB
<b>Velocidad reloj</b>	16 Mz

Realizado por: JARA, Caroline; CABA, Carlos 2018

### 1.3.4 Cerradura Eléctrica



**Figura 22-1:** Cerradura eléctrica.  
Fuente: (Thayer, 2018).

Una cerradura eléctrica estándar es muy utilizada porque puede abrirse a distancia con solo pulsar un interruptor, funciona con 12 voltios por lo que utiliza un transformador. Opera del siguiente modo, cuando se pulsa el interruptor, la corriente pasa por el primario del transformador a 110 voltios, el secundario del mismo a 12 voltios conectado en serie con el solenoide de la cerradura hace que esta se magnetice desactivando el enclavamiento que asegura el pestillo más ancho de la cerradura, el mismo que recorre a la posición de abierto gracias a la acción de un resorte que esta tensado el momento que se cierra la cerradura. (mecánica, 2014)

### 1.3.5 Hub USB



**Figura 23-1:** Conector Hub USB.  
Fuente: (Facil, 2014)

El hub USB posibilita la entrada adicional de elementos que tengan conexión USB con nuestro procesador, estos son las web cam y arduino.

Un hub USB es un dispositivo que permite concentrar varios puertos USB, permitiendo la conexión con una máquina mediante un solo bus o cable. Los hub USB se integran a menudo en la propia computadora, en teclados o, más raramente, en monitores o impresoras. Los hub USB vienen en una variedad amplia de formas: similares a un concentrador, diseños pequeños previstos para ser conectado directamente en el puerto USB de la computadora, etc. (Facil, 2014)

### 1.3.6 Adaptador de corriente



**Figura 24-1:** Adaptador de Corriente  
Fuente: (Facil, 2014)

Los adaptadores de corriente alterna se utilizan normalmente con los dispositivos eléctricos que no contienen su propia fuente de alimentación interna. Los circuitos internos de una fuente de alimentación externa son muy similares en diseño al que se utiliza para la alimentación imbuido (built-in) o interna, pero existen varias ventajas de separar la fuente de alimentación del cuerpo principal del dispositivo electrónico.

### 1.3.7 Elementos electrónicos.

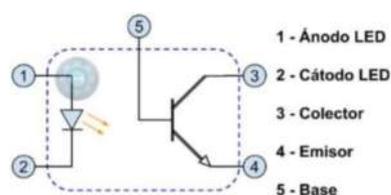
#### 1.3.7.1 Resistencia



**Figura 25-1:** Resistencia  
Fuente: (Thayer, 2018).

Resistencia eléctrica es toda oposición que encuentra la corriente a su paso por un circuito eléctrico cerrado, atenuando o frenando el libre flujo de circulación de las cargas eléctricas o electrones. Cualquier dispositivo o consumidor conectado a un circuito eléctrico representa en sí una carga, resistencia u obstáculo para la circulación de la corriente eléctrica.

#### 1.3.7.2 Optoacoplador



**Figura 26-1:** Optoacoplador  
Fuente: (Thayer, 2018).

Es un circuito electrónico que funciona como un interruptor aislado ópticamente. Es decir, que permite una conexión eléctricamente aislada entre dos circuitos que operan a distintos voltajes. Está construido por un led y un circuito de control activado por luz infrarroja. Entre otras cosas, una de las ventajas principales de los opto-acopladores es su aislación eléctrica entre la carga y la electrónica de control. La única conexión entre ambos elementos es la luz del led que activa al foto-transistor.

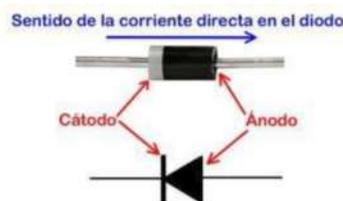
### 1.3.7.3 Relé



**Figura 27-1:** Relé  
Fuente: (Thayer, 2018).

Es un dispositivo que consta de dos circuitos diferentes: un circuito electromagnético (electroimán) y un circuito de contactos, al cual aplicaremos el circuito que queremos controlar. Su funcionamiento se basa en el fenómeno electromagnético. Cuando la corriente atraviesa la bobina, produce un campo magnético que magnetiza un núcleo de hierro dulce (ferrita). Este atrae al inducido que fuerza a los contactos a tocarse. Cuando la corriente se desconecta vuelven a separarse.

### 1.3.7.4 Diodo



**Figura 28-1:** Diodo  
Fuente: (Thayer, 2018).

Los diodos son dispositivos semiconductores que permiten hacer fluir la electricidad solo en un sentido. La flecha del símbolo del diodo muestra la dirección en la cual puede fluir la corriente.

Los diodos son la versión eléctrica de la válvula o tubo de vacío y al principio los diodos fueron llamados realmente válvulas.

#### 1.3.7.5 *Transistor*



**Figura 29-1:** Transistor

**Fuente:** (Thayer, 2018).

Un transistor es un dispositivo que regula el flujo de corriente o de tensión actuando como un interruptor o amplificador para señales electrónicas. Es un componente electrónico formado por materiales semiconductores, de uso muy habitual, pues lo encontramos presente en cualquiera de los aparatos de uso cotidiano como las radios, alarmas, automóviles, ordenadores, etc.

## 1.4 Software

### 1.4.1 *Matlab*



**Figura 30-1:** Logo Matlab

**Fuente:** <https://engineering.usu.edu/images/logos-wordmarks/logos/MATLAB-Logo.png>

MATLAB (abreviatura de “Matrix Laboratory”, o laboratorio de matrices) fue creado por Cleve Moler en 1984, es un lenguaje de alto nivel de Mathworks con un entorno de desarrollo integrado propio que permite realizar funciones de cálculo numérico, visualización y programación en análisis de datos, desarrollo de algoritmos y creación de modelos o aplicaciones.

#### 1.4.1.1 *Características*

En MATLAB la programación sigue una estructura similar a los lenguajes C, C++ o Java, y es ampliamente utilizado en sectores tan dispares como el industrial, el económico-financiero, el educativo, etc.

Además, el lenguaje del cálculo técnico de MATLAB se utiliza en gran cantidad de universidades y centros de investigación de todo el mundo. Sus funciones básicas son la manipulación de matrices, la representación de datos y funciones, la implementación de algoritmos, la creación de interfaces de usuario (GUI) y la comunicación con programas en otros lenguajes y con otros dispositivos hardware.

Tiene la capacidad de ampliar sus funcionalidades mediante la adición de “toolboxes” por instalación de paquetes propios de la compañía Mathworks. Algunas de estas “toolboxes” son: visión por computador, adquisición de datos, redes neuronales, robótica, adquisición y procesamiento de imagen.

#### 1.4.1.2 *Comandos y funciones de Matlab*

Para crear un programa en Matlab, hay que usar el Editor y guardarlo en el directorio correspondiente. Hay dos tipos de programas, útiles dependiendo de la dificultad requerida:

**Scripts:** Son programas útiles para unir una serie de líneas ejecutables simples y mecánicas. Consisten en una sucesión de comandos, que se ejecutan en el Command Window de principio a fin. Para crear el script, se escriben los datos de entrada en el Editor de esta manera:

**Funciones:** Son programas que realizan tareas más elaboradas que, a partir de ciertos argumentos de entrada, realizan una serie de cálculos programados en el Editor, con una serie de líneas ejecutables y devuelven uno o varios argumentos de salida al llamar la función desde el Command Window. La primera línea de la función debe estar escrita de la siguiente manera:

En este proyecto se aplica Matlab para el desarrollo del código y adquisición de datos para el reconocimiento rostros y creación del modelo facial 3D ya que es un software muy potente y es empleado en los grandes centros de investigación a nivel mundial.

### 1.4.2 *MySQL server*



**Figura 31-1:** Logo MySQL

**Fuente:** <https://www.linuxadictos.com/wp-content/uploads/1200px-MySQL.svg.png>

El software MySQL se utiliza para elaborar y gestionar la base de datos que se ocupa en el sistema.

MySQL, es un sistema de gestión de base de datos relacional. El software MySQL® proporciona un servidor de base de datos SQL (Structured Query Language) muy rápido, multi-threaded, multi usuario y robusto que le permite ser utilizado por varias personas al mismo tiempo y realizar varias consultas a la vez, lo que lo hace sumamente versátil.

El servidor MySQL está diseñado para entornos de producción críticos, con alta carga de trabajo así como para integrarse en software para ser distribuido. MySQL es una marca registrada de MySQL AB.

La mayor parte del código se encuentra escrito en lenguaje C/C++ y la sintaxis de su uso es bastante simple, lo que permite crear bases de datos simples o complejas con mucha facilidad. Además, es compatible con múltiples plataformas informáticas y ofrece una infinidad de aplicaciones que permiten acceder rápidamente a las sentencias del gestor de base de datos.

La clave de su éxito es que se trata de un sistema de libre distribución y de código abierto. Lo primero significa que se puede descargar libremente de Internet (por ejemplo de la dirección ([www.mysql.com](http://www.mysql.com))); lo segundo (código abierto) significa que cualquier programador puede remodelar el código de la aplicación para mejorarlo. Esa es también la base del funcionamiento del sistema Linux, por eso MySQL se distribuye fundamentalmente para Linux, aunque también hay versiones para Windows. (Sánchez, 2003, p. 2).

## CAPITULO II

### 2 MARCO METODOLÓGICO

#### 2.1 Proceso de elaboración del proyecto

##### 2.1.1 *Investigación de la información*

Se realiza una investigación de la información referente a reconocimiento por el método EBGM y creación del modelo facial por el método 3DMM, dicha investigación abarca todas sus definiciones, pasos constitutivos, ecuaciones, modelos, etc. Se recopila los datos más importantes y se analizan con el fin de tener una mejor comprensión del tema y así lograr desarrollar el proyecto.

##### 2.1.2 *Elección de los elementos a utilizar*

Contando con toda la información referente al tema se realiza una investigación dedicada a que elemento se debe utilizar, que elementos existen, cuales son las características de los elementos que se adaptan de mejor forma al proyecto. En este caso se decide realizar un sistema que cuenta con un computador portátil como unidad de procesamiento, un par de cámaras web como elemento de ingreso de información, una placa arduino y una placa electrónica la cual se elabora como intermediario entre el procesador y los elementos de ingreso de información, una cerradura eléctrica que es el dispositivo de salida.

En cuanto a la parte de software se programa en el entorno Matlab lo referente a algoritmos de reconocimiento facial y modelo facial 3D, en MySQL la gestión de base de datos y registro de usuarios, además se controla la placa arduino dentro del mismo entorno en el que se desarrolla el sistema y se decide utilizar el software también libre denominado Eagle para la elaboración de la placa necesaria. La comunicación entre Matlab y MySQL se la ejecuta usando el conector ODBC.

##### 2.1.3 *Verificación de compatibilidad entre elementos.*

Se establece que la compatibilidad entre software y hardware elegidos si es posible con tan solo realizar unas instalaciones complementarias de paquetes, además se aprueba la comunicación

entre los dispositivos de ingreso de información y la placa elaborada con el fin de controlar el dispositivo de salida (cerradura eléctrica).

#### **2.1.4 *Elaboración del software.***

Teniendo a disposición todos los elementos y software que se va a utilizar se elabora la programación que abarca lo relacionado a procesamiento de imágenes (utilizado en la captura de fotografías y su procesamiento), detección facial, conexión y gestión a base de datos, conexión con arduino, desarrollo de interface gráfica.

#### **2.1.5 *Descripción del lugar de aplicación.***

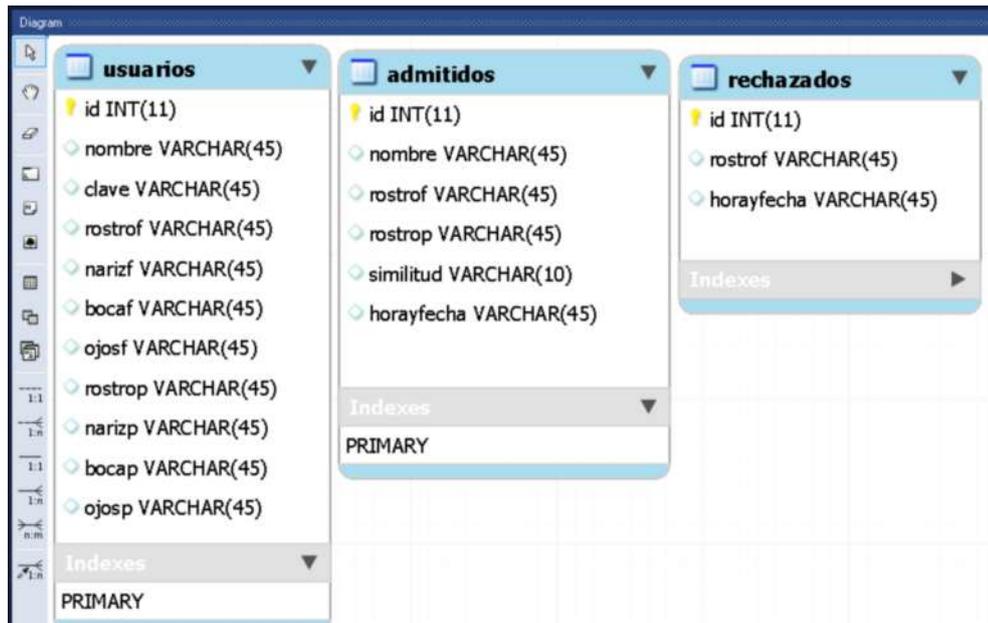
El proyecto se elabora con la finalidad de utilizarlo en un área o lugar que requiera un nivel de seguridad apropiado y más sofisticado, estos pueden ser ingreso a laboratorios privados, oficinas administrativas, depósitos de materiales, domicilios, etc. Para el caso de éste proyecto en específico se lo instala en un domicilio que controla el acceso a un cuarto de estudio.

#### **2.1.6 *Proceso de pruebas y planteamiento de resultados.***

El método que se utiliza para la elaboración de los resultados es el método experimental que consta de un conjunto de nueve pruebas bajo distintas condiciones y se verifica como responde el sistema ante cada una de ellas, para la toma de datos en la elaboración de resultados se utiliza el método de observación experimental debido a que el sistema se implementa en un lugar de fácil acceso para la toma de información.

## **2.2 Estructura de la base de datos**

Antes se había mencionado que la información de los usuarios tanto de admitidos y rechazados se encuentra almacenada en la base de datos elaborada en MySQL, esta base de datos cuenta con tres tablas principales, tabla usuario, tabla admitidos y tabla rechazados. En la sección anterior ya se describió como está compuesta cada una de las tablas, ahora mostraremos cual es la estructura general de la base.



**Figura 1-2:** Tablas de la base de datos del proyecto  
 Realizado por: JARA, Caroline; CABA, Carlos 2018.

Como se observa en la figura 1-2 la estructura de la base de datos no muestra ningún tipo de conexión entre tablas o más conocida como dependencia entre tablas, esto se debe a que se realiza fuera del entorno MySQL todo lo referente a relación o interacción entre cada una de las tablas.

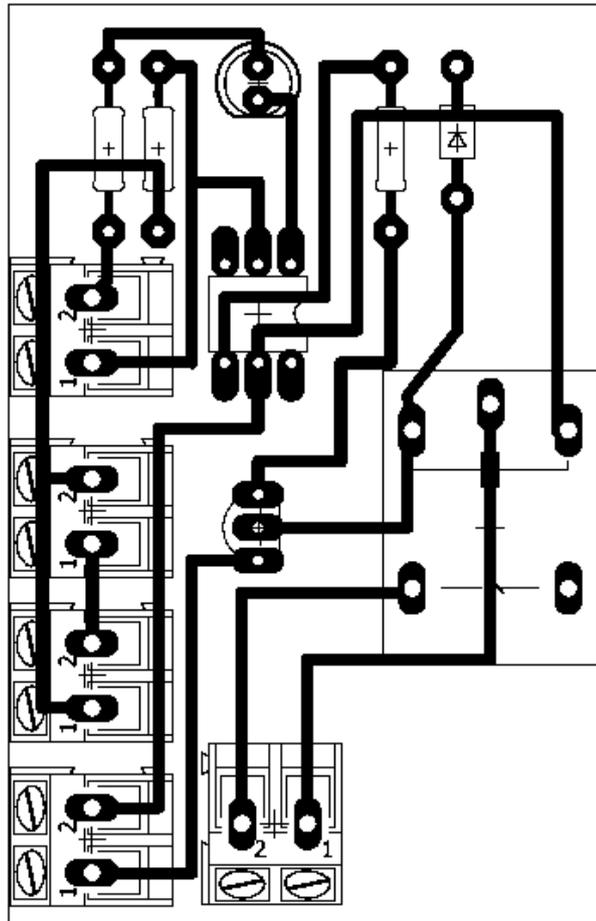
### 2.3 Elaboración de la placa de conexión entre Matlab y los dispositivos.

En este apartado se describe cual fue el proceso de elaboración de la placa que hemos utilizado y también su funcionamiento.

#### 2.3.1 *Diseño de la placa*

El programa que se usa para el diseño de pistas para el circuito es Eagle 7.6.0, debido a que es un programa conocido por los autores y de agradable interface para los usuarios.

A continuación en la figura 2-2 se muestra el diseño de las pistas con una herramienta que nos facilita el programa Eagle.

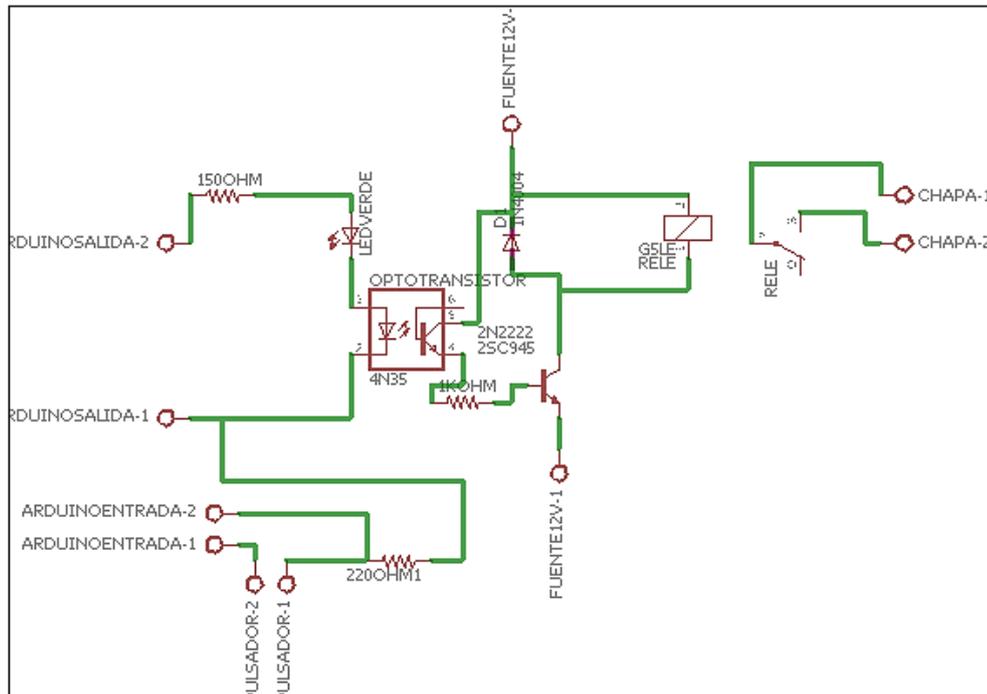


**Figura 2-2:** Diseño de pistas del circuito  
 Realizado por: JARA, Caroline; CABA, Carlos 2018

### 2.3.2 Principio de funcionamiento

El circuito que se ha diseñado cumple con dos trabajos primordiales, el primero aísla a nuestro controlador (Arduino) del circuito de mando que funciona con 12v DC para evitar posibles fallos por sobre voltajes y también como medida de seguridad contra posibles malos funcionamientos de los elementos.

Como segundo trabajo que cumple es acoplar perfectamente a una salida del Arduino con el control total de la chapa eléctrica, esto lo realiza mediante el uso de un circuito de control ON/OFF por medio de un transistor y un relé, para una mejor comprensión se muestra en la figura 3-2 el diseño del circuito.



**Figura 3-2:** Circuito de control de actuadores  
**Realizado por:** JARA, Caroline; CABA, Carlos 2018.

Ahora bien el circuito se puede dividirlo en tres partes: La primera conformada por las borneras denominadas *ARDUINOSALIDA-1* y *ARDUINOSALIDA-2*, la resistencia de *150OHM*, el *LEDVERDE* y el opto transistor *4N35* tiene como finalidad aislar por completo la parte de mando que funciona con 12V DC de las entradas/salidas del controlador que trabajan con 5V DC y también acoplan su acción de tal manera que se pueda controlar la acción de la chapa eléctrica con una salida digital del Arduino.

La segunda parte del circuito se conforma por las borneras *ARDUINOENTRADA-1*, *ARDUINOENTRADA-2*, *PULSADOR-1*, *PULSADOR-2*, y la resistencia de *220OHM* tienen como finalidad detectar cuando el usuario solicita acceso al sistema y a su vez también proporcionan de alimentación de los 5V DC para el funcionamiento de la primera parte del circuito.

Por ultimo tenemos la denominada “parte de mando” que lo conforman la resistencia de *1kOHM*, el transistor *2N2222*, el diodo *1N4004*, el relé de 12V DC y las borneras denominadas *FUENTE12V-1*, *FUENTE12V-2*, *CHAPA-1*, *CHAPA-2* cumplen con la función de tomar una señal de entrada proveniente del arduino por medio del opto acoplador en la base del transistor que está configurado como interruptor y permitir el paso de corriente a través del relé lo cual activará sus contactos y logrará energizar la chapa eléctrica que admite como voltaje de

activación 12V DC, adicional por un par de sus borneras será donde se suministre la fuente de 12V DC a 3A.

La razón por la se decidió usar esa fuente es debido a que la chapa eléctrica tiene un consumo aproximado de 1.75 A y también porque admite ya sea voltaje AC como DC.

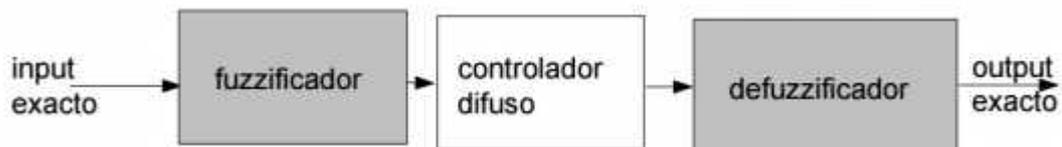
## 2.4 Algoritmo de inteligencia artificial

Esta sección se divide en dos partes, en la primera parte se va a explicar bien a detalle lo correspondiente al método de lógica difusa que hemos utilizado, para ello nos apoyamos en una herramienta de Matlab denominada FLD que por sus siglas en ingles significa “Diseñador de Lógica Difusa” (“Fuzzy Logic Designer”). Aquí hablaremos de cuáles son las variables que se toman como entrada, de la configuración de la Lógica Difusa que se utiliza y de los resultados que se obtiene.

Luego en la segunda parte se centra más a detalle en cómo se incluye el método utilizado en el programa en sí y para ello se describe paso a paso el proceso de reconocimiento en su totalidad.

### 2.4.1 Descripción del algoritmo de Lógica Difusa

La lógica difusa posee una estructura básica definida como se muestra en la siguiente figura 4-2



**Figura 4-2:** Estructura de algoritmo de Lógica Difusa  
Realizado por: JARA, Caroline; CABA, Carlos 2018

Entonces antes de aplicar el método difuso primero se va a configurar de acuerdo a las necesidades para lo cual vamos a definir ciertos parámetros como son:

- **Objetivo:** Se necesita elaborar un método que sea capaz de verificar si una fotografía de una persona tiene o no similitud con alguna de las fotografías almacenadas en la base de datos.
- **Parámetros previamente definidos:** Se ha decidido que para nuestro método difuso se va a utilizar un controlador difuso de tipo Mamdani ya que éste nos brinda como salida un valor lógico que para nuestra aplicación sería si coincide o no con la persona. También

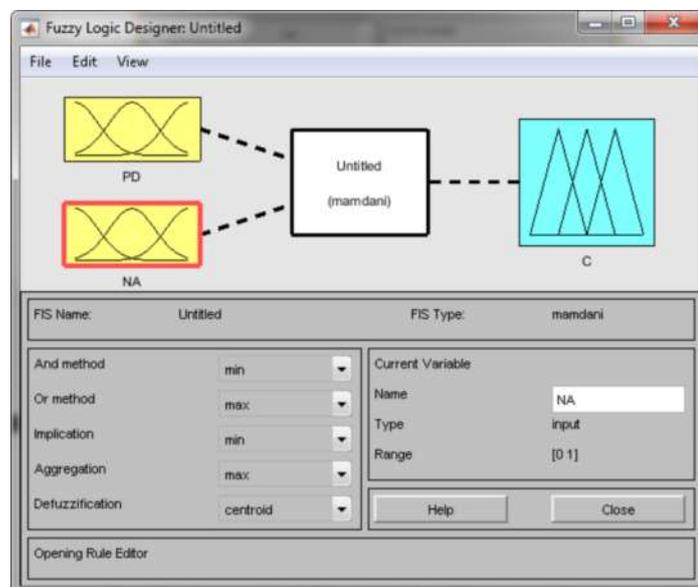
hemos optado por utilizar un defuzzificador de tipo centroide ya que brinda un promedio confiable en la toma de decisiones. (Jiménez, 2013)

- **Supuestos:** Se tiene claro que para la elaboración del método se está seguro de que no existirá otra condición adicional a las que utilizemos, es decir que abarcamos todas las posibilidades de entradas.
- **Forma de función de pertenencia:** Se utiliza la función de pertenencia de forma trapezoidal debido a que facilita el trabajo a la hora de elaborar las funciones de membresía.

Ahora ya conociendo los parámetros mencionados se procede a la elaboración del método para lo cual tenemos:

### 1. Obtención del modelo.

Para el objetivo planteado se utiliza únicamente dos variables de entrada las cuales son: “PD” (Posiciones detectadas) y “NA” (Numero de aciertos), y con una sola variable de salida “C” (Coincidente). A continuación en la figura 5-2 se muestra el modelo planteado para el problema.



**Figura 5-2:** Modelo de Lógica Difusa  
Realizado por: JARA, Caroline; CABA, Carlos 2018

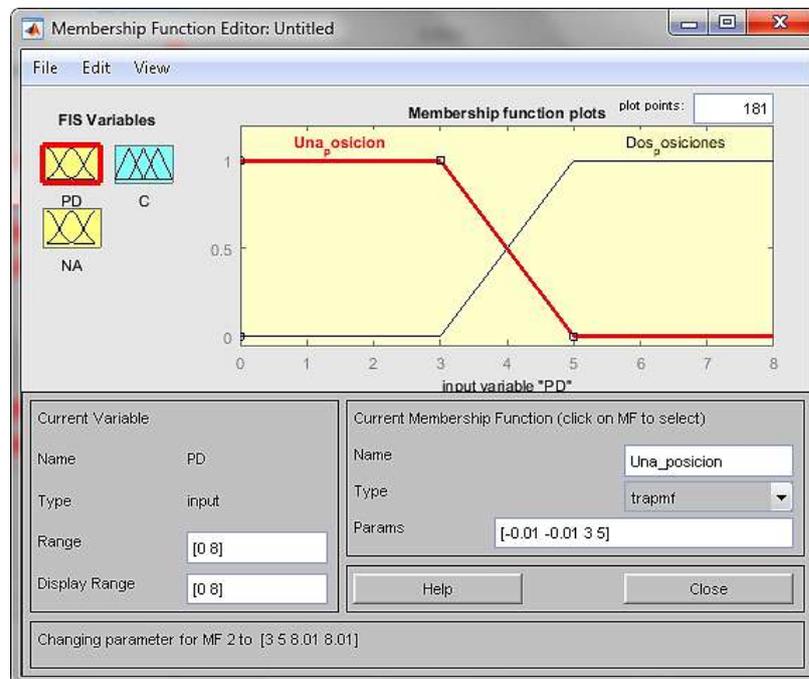
### 2. Creación de las variables.

La variable “PD” como se puede ver en la figura 6-2 es una variable de entrada que se refiere a las posiciones detectadas en la imagen, es decir en nuestro caso puede haber solo detectada la posición de frente, o solo detectada la posición de perfil, o ambas. Se ha decidido darle una escala de 0 a 8 que representa el total de imágenes que se tendría si hubiese detectado ambas posiciones y se dividió a la variable de la siguiente forma:

0-5: Una posición.

4-8: Dos posiciones

Nota: La opción de que no se detecte ninguna posición no está contemplada ya que se tiene el control previo en el programa de que exista al menos una posición para avanzar a este algoritmo.



**Figura 6-2: Variable PD**

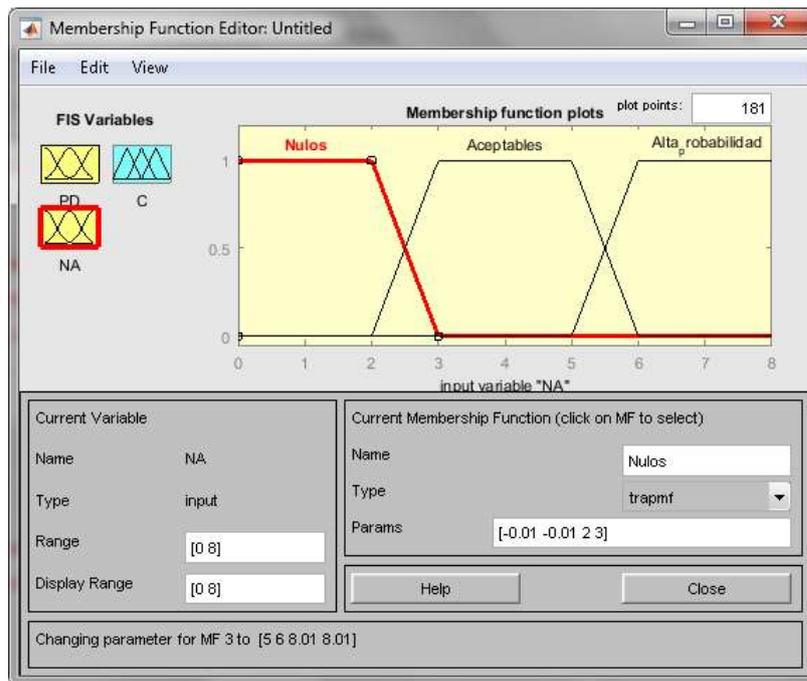
Realizado por: JARA, Caroline; CABA, Carlos 2018

La variable “NA” como se puede ver en la figura 7-2 es una variable de entrada que hace referencia al número de aciertos que presentó el usuario activo con los usuarios registrados en la base de datos. Igualmente se dio una escala de 0 a 8 ya que es el número máximo de aciertos posibles que puede tener la persona y se dividió a la variable de la siguiente forma:

0-3: Nulos

2-6: Aceptables

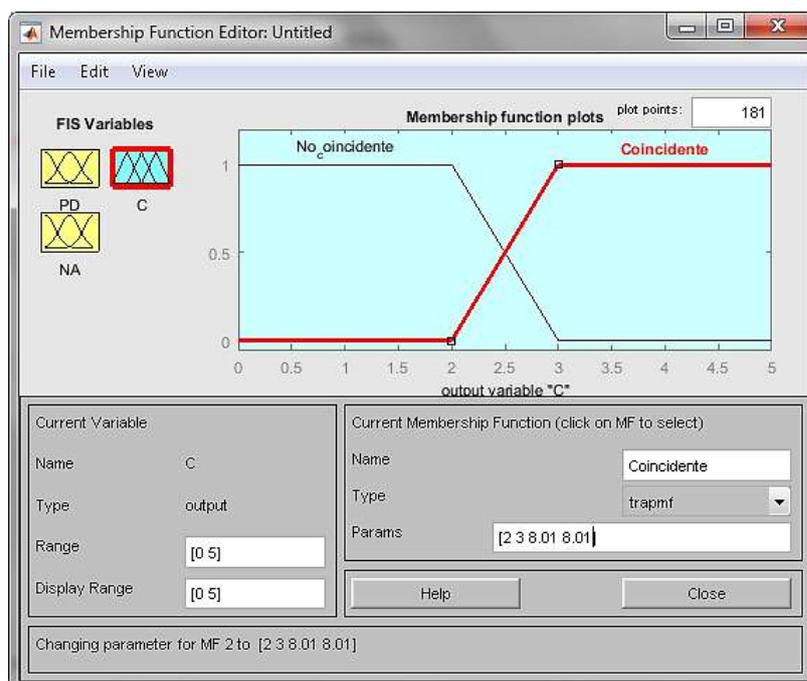
5-8: Alta probabilidad



**Figura 7-2: Variable NA**  
 Realizado por: JARA, Caroline; CABA, Carlos 2018

La variable “C” como se puede ver en la figura 8-2 es una variable de salida que indicará si el usuario activo ha sido coincidente o no con el usuario de la base de datos. Su escala esta de 0 a 5 en donde se tiene las siguientes posibilidades:

- 0-3: No coincidente
- 2-5: Coincidente

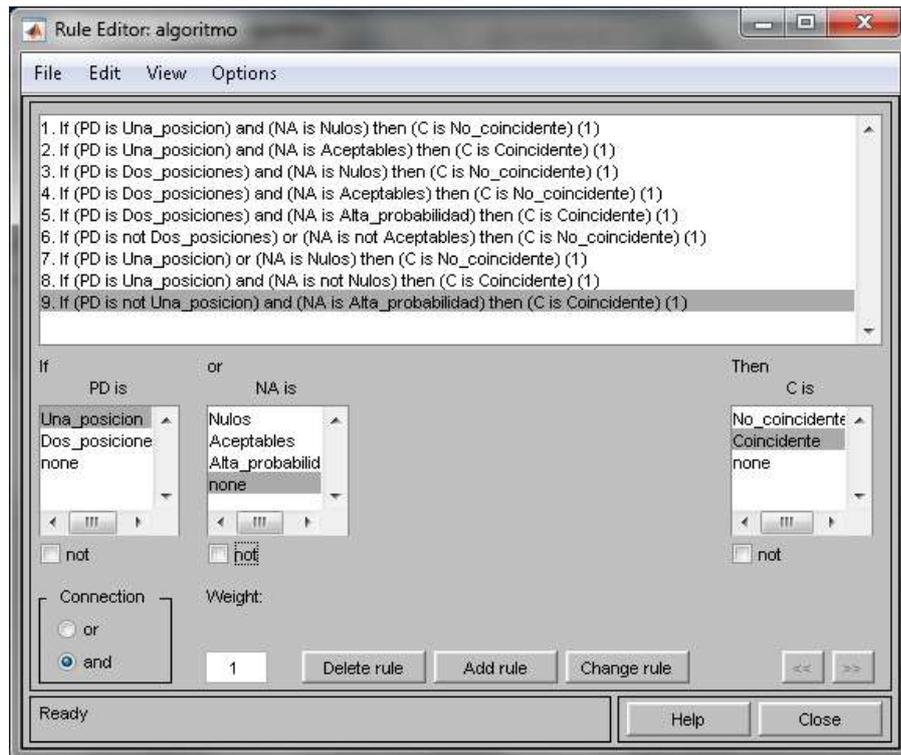


**Figura 8-2: Variable C**  
 Realizado por: JARA, Caroline; CABA, Carlos 2018

### 3. Creación de las reglas fuzzy

Debido a que se utiliza dos variables de ingreso y que tienen dos y tres funciones de membresía respectivamente el número total de reglas fuzzy que se debe utilizar son seis reglas.

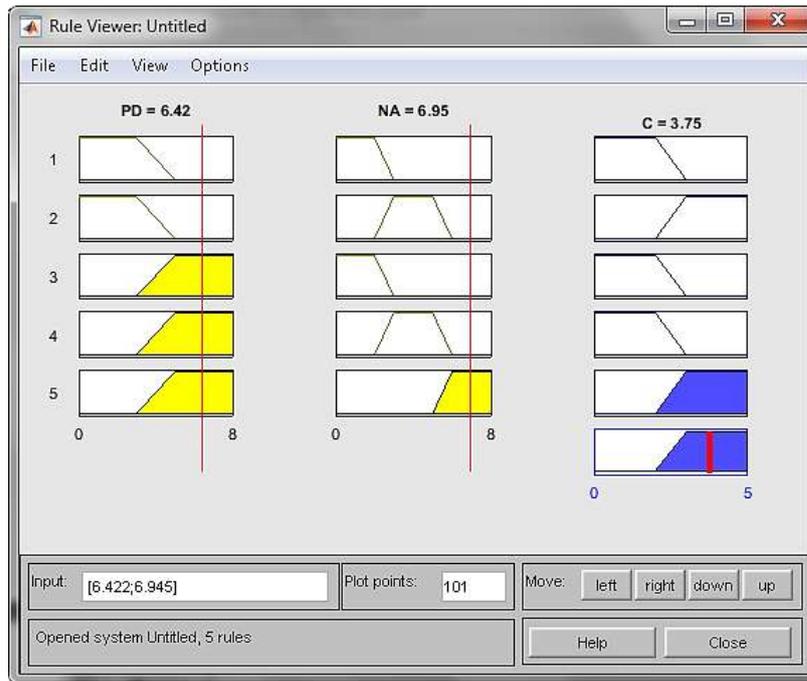
Pero como se considera la utilización de relación de unión, intersección y negación el número de reglas que se aplica para controlar todos los posibles casos son nueve. En la figura 9-2 se puede ver las reglas fuzzy ocupadas en el sistema.



**Figura 9-2:** Reglas fuzzy del sistema  
Realizado por: JARA, Caroline; CABA, Carlos 2018

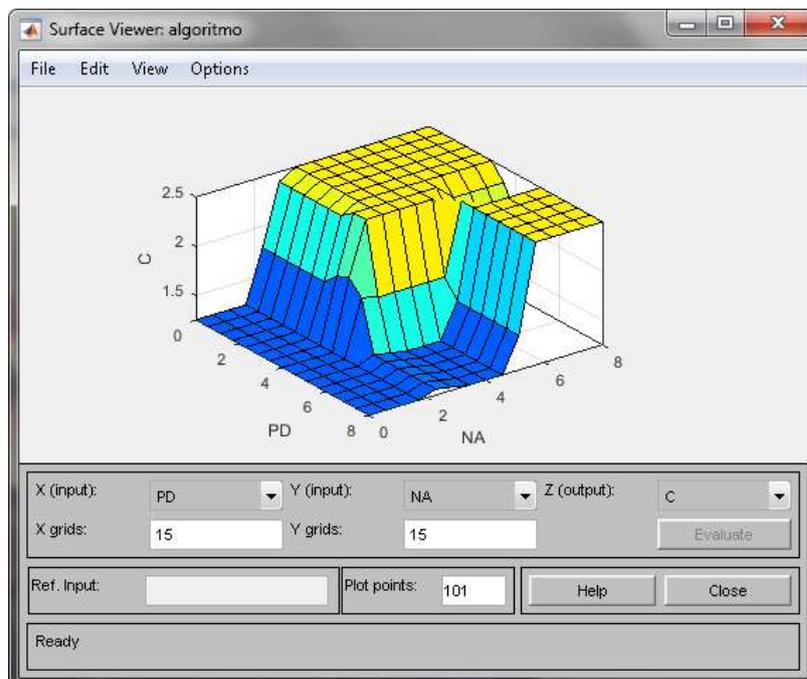
### 4. Visualización de resultados

La herramienta que se utiliza para poder presentar de forma dinámica el método difuso que se utiliza nos permite mostrar los resultados de dos formas distintas. En la primera forma mostrada en la figura 10-2 podemos ver cómo podemos simular las entradas que puedan darse moviendo la línea roja y nos dará una respuesta indicada en la parte derecha del dibujo en azul.



**Figura 10-2:** Resultados fuzzy forma dinámica  
**Realizado por:** JARA, Caroline; CABA, Carlos 2018

En cuanto a la segunda forma en que nos muestra los resultados como podemos ver en la figura 11-2 está representado en 3D en donde se aprecia a simple vista dos niveles que son el nivel pintado en azul que representa cuando el resultado es no coincidente y el nivel representado en amarillo que abarca las combinaciones en las cuales el resultado es coincidente.



**Figura 11-2:** Resultados fuzzy forma 3D  
**Realizado por:** JARA, Caroline; CABA, Carlos 2018

## 2.4.2 Inclusión del algoritmo implementado en el sistema

El método de lógica difusa que se describió anteriormente se lo ha incluido en el sistema y el funcionamiento total del mismo se describe a continuación en la figura 12-2.



**Figura 12-2:** Algoritmo de Inteligencia Artificial  
Realizado por: JARA, Caroline; CABA, Carlos 2018

### 2.4.2.1 Obtención de los datos de cada usuario

Para poder establecer una lógica difusa se debe poseer nuestro conjunto de datos que será tratado como el universo de trabajo. Este conjunto de datos no será más que el resultado de aplicar los filtros de Gabor a cada uno de los usuarios de la base de datos. A continuación en la figura 13-2 mostramos el código correspondiente y su explicación.

```
[rostrope,narizpe,bocape,ojospe]=procesarimagen(imgp);

input_workrostrop=obtenervectorescaracteristicos(rostrope);
input_worknarizp=obtenervectorescaracteristicos(narizpe);
input_workbocap=obtenervectorescaracteristicos(bocape);
input_workojosp=obtenervectorescaracteristicos(ojospe);

similarity_scorerostrop=arrayfun(@(n)1/(norm(featuresrostrop(:,n)-feature_vecrostrop)),1:usuarios);
similarity_scorenarizp=arrayfun(@(n)1/(norm(featuresnarizp(:,n)-feature_vecnarizp)),1:usuarios);
similarity_scorebocap=arrayfun(@(n)1/(norm(featuresbocap(:,n)-feature_vecbocap)),1:usuarios);
similarity_scoreojosp=arrayfun(@(n)1/(norm(featuresojosp(:,n)-feature_vecojosp)),1:usuarios);
```

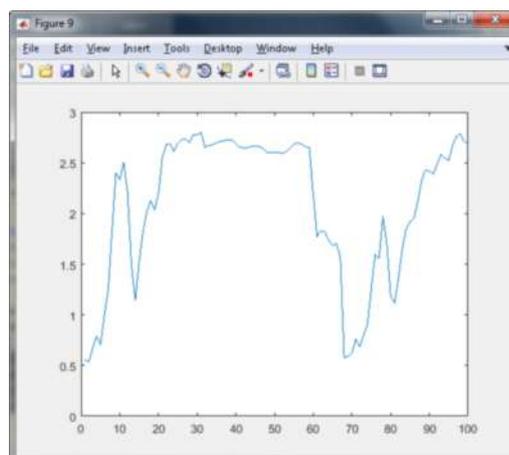
**Figura 13-2:** Código de obtención banco de filtros de los usuarios  
Realizado por: JARA, Caroline; CABA, Carlos 2018.

Como se puede ver en la Figura 13-2 el primer paso es llamar a nuestra función procesarimagen que ejecutará lo explicado en la sección de visión artificial en lo cual recortará la imagen tal como necesitamos para nuestro algoritmo. Luego se ejecuta la función la cual aplica el filtro de Gabor a cada imagen y nos dará como resultado el banco de filtros de Gabor, más adelante se detallará con más claridad que funciones se utilizaron para lograr lo comentado. Por último se aplica una comparación entre cada filtro utilizando un método de comparación por la distancia de Euler y se almacenara todos los resultados en una misma matriz para luego proceder a tomar la decisión si es o no el usuario registrado.

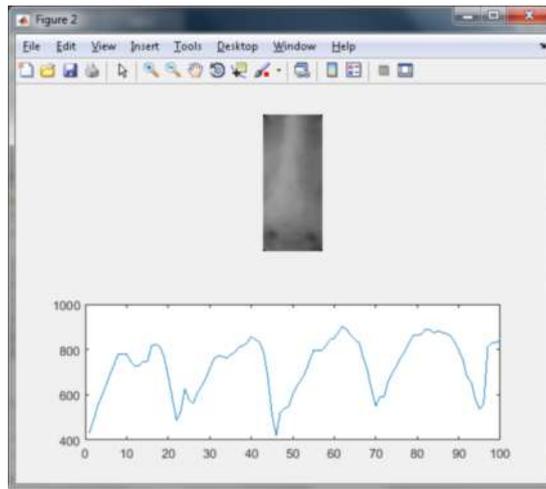
A continuación en las figuras 14-2 hasta la figura 23-2 se mostrará un ejemplo de los vectores característicos obtenidos de la fotografía de frente y perfil de uno de los usuarios con sus respectivas partes.



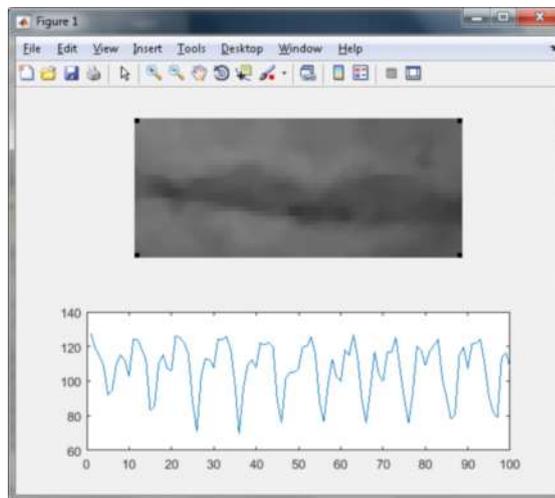
**Figura 14-2:** Rostro de frente del usuario  
Realizado por: JARA, Caroline; CABA, Carlos2018



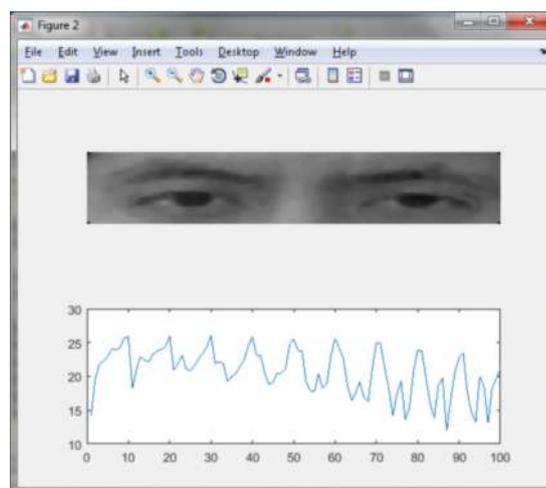
**Figura 15-2:** Vectores característicos rostro  
Realizado por: JARA, Caroline; CABA, Carlos 2018



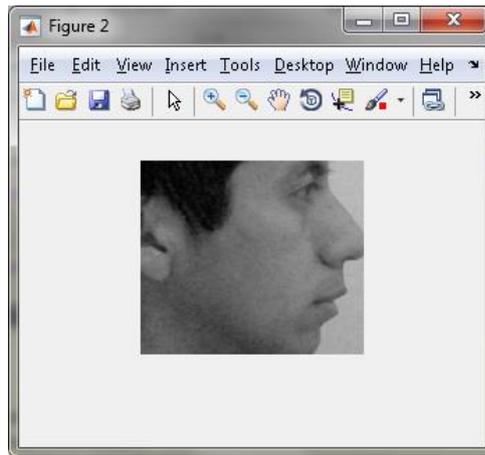
**Figura 16-2:** Vectores característicos nariz de frente  
 Realizado por: JARA, Caroline; CABA, Carlos 2018



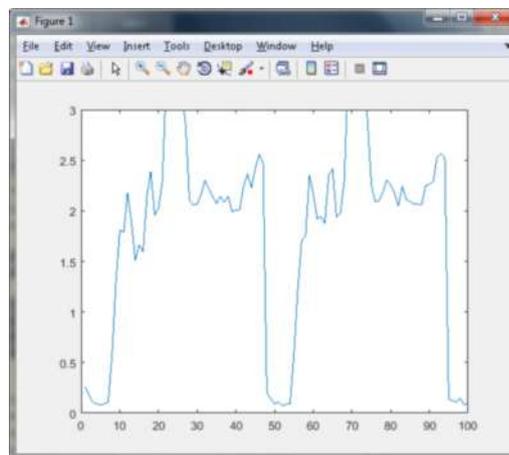
**Figura 17-2:** Vectores característicos boca de frente  
 Realizado por: JARA, Caroline; CABA, Carlos 2018



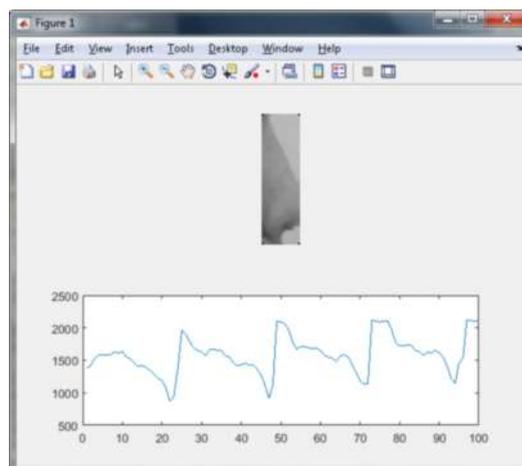
**Figura 18-2:** Vectores característicos ojos de frente  
 Realizado por: JARA, Caroline; CABA, Carlos 2018



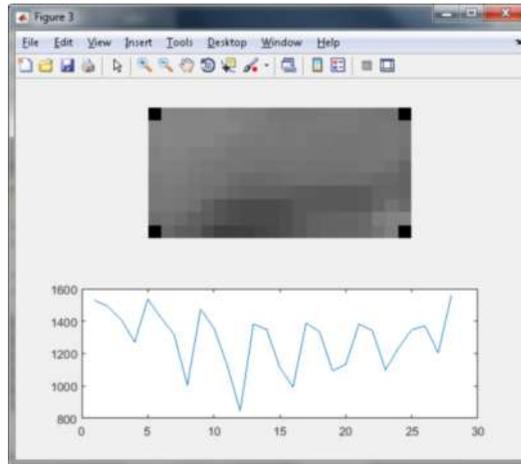
**Figura 19-2:** Rostro de perfil del usuario  
 Realizado por: JARA, Caroline; CABA, Carlos2018



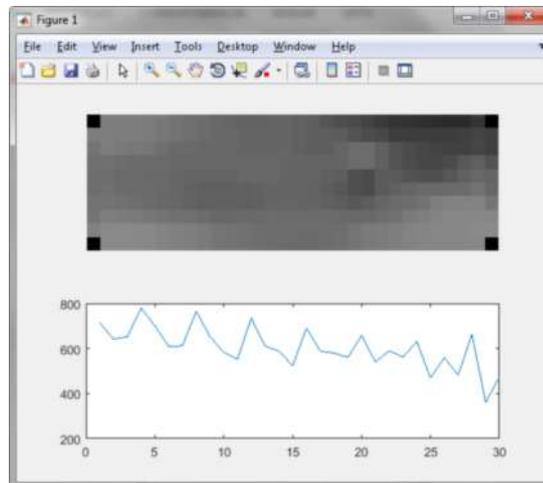
**Figura 20-2:** Vectores característicos rostro perfil  
 Realizado por: JARA, Caroline; CABA, Carlos 2018



**Figura 21-2:** Vectores característicos nariz de perfil  
 Realizado por: JARA, Caroline; CABA, Carlos 2018



**Figura 22-2:** Vectores característicos boca de perfil  
**Realizado por:** JARA, Caroline; CABA, Carlos 2018



**Figura 23-2:** Vectores característicos ojos de perfil  
**Realizado por:** JARA, Caroline; CABA, Carlos 2018

#### 2.4.2.2 Definir cuál es la mayor coincidencia en relación a las demás

Para el siguiente paso se determina cuáles fueron las mejores coincidencias en cada uno de los campos que se está analizando como son, rostro total, nariz, boca, y ojos tanto de posición frontal como perfilada. Se entiende que mejor coincidencia no corresponde a un valor establecido fijo de similitud ya que como vimos los vectores toman escalas diferentes para cada una de las partes analizadas, entonces para poder verificar si hay un ganador o no se clasifica el valor más alto de todos los usuarios como el correspondiente al usuario ganador y luego se procederá a tomar alguna decisión si se acepta o no al usuario.

A continuación se muestra un ejemplo de comparaciones de similitud con cuatro usuarios registrados en la base de datos.

```

similarity_scorerostrof =
    1.8239    1.5630    1.5820    4.0404

similarity_scorenarizf =
    0.2589    0.0987    0.2421    0.4675

similarity_scorebocaf =
    2.3745    0.6866    1.0723    1.8055

similarity_scoreojosf =
    4.6249    2.5680    3.6202    20.8667

```

**Figura 24-2:** Porcentajes de similitud con cuatro usuarios registrado en la base de datos (frente)  
**Realizado por:** JARA, Caroline; CABA, Carlos 2018

```

similarity_scorerostrop =
    1.6955    1.5719    1.7008    2.7306

similarity_scorenarizp =
    0.0793    0.1263    0.1286    0.3818

similarity_scorebocap =
    0.2896    0.2943    0.2343    0.4086

similarity_scoreojosp =
    0.6005    0.4803    1.1295    0.8361

```

**Figura 25-2:** Porcentajes de similitud con cuatro usuarios registrado en la base de datos (perfil)  
**Realizado por:** JARA, Caroline; CABA, Carlos 2018

Como se observa en este caso en particular la figura 24-2 describe los porcentajes de similitud de los cuatro usuarios registrado en la base de datos en comparación con el usuario que ha solicitado el ingreso y que está siendo procesado. En la figura 25-2 se tiene también los porcentajes pero en la comparación que se realiza con la entrada tomada de perfil. Aquí se observa que el ganador en la mayoría de campos es el usuario número cuatro, pero no es necesario únicamente con tener el mayor porcentaje de similitud en alguno de los campos hay que cumplir con las condiciones establecidas en el algoritmo de inteligencia artificial para considerarlo como usuario coincidente. En el siguiente punto se explicará más a detalle la forma de identificar si un usuario seleccionado es correcto o no.

### 2.4.2.3 Determinar si el usuario seleccionado es correcto o no

Una vez que se obtiene todos los bancos de vectores de Gabor de cada usuario el siguiente punto será verificar si existe o no un usuario ganador y para ello se basa en la definición que hemos dado de lógica difusa, ahora como ya tenemos el universo de trabajo nos faltaría ver la diferencia relativa entre cada uno de los usuarios y para ello compararemos cada rostro, cada parte del rostro (nariz, boca y ojos) y se ve si es en realidad la persona registrada al menos tendrá la diferencia más alta en la mayoría de comparaciones que los demás usuarios.

Para una mejor comprensión a continuación se muestra el código correspondiente a lo descrito con su respectiva explicación.

```
suma=zeros(1,usuarios);

prf=find(vrf==max(vrf));
suma(1,prf)=suma(1,prf)+1;
pnf=find(vnf==max(vnf));
suma(1,pnf)=suma(1,pnf)+1;
pbf=find(vbf==max(vbf));
suma(1,pbf)=suma(1,pbf)+1;
pof=find(vof==max(vof));
suma(1,pof)=suma(1,pof)+1;

prp=find(vrp==max(vrp));
suma(1,prp)=suma(1,prp)+1;
pnp=find(vnp==max(vnp));
suma(1,pnp)=suma(1,pnp)+1;
pbp=find(vbp==max(vbp));
suma(1,pbp)=suma(1,pbp)+1;
pop=find(vop==max(vop));
suma(1,pop)=suma(1,pop)+1;

suma1=suma;

maximo1=max(suma1);
pmaximo1=find(suma1==max(suma1));
suma1(1,pmaximo1)=0;
maximo2=max(suma1);
diferencia=maximo1-maximo2;
```

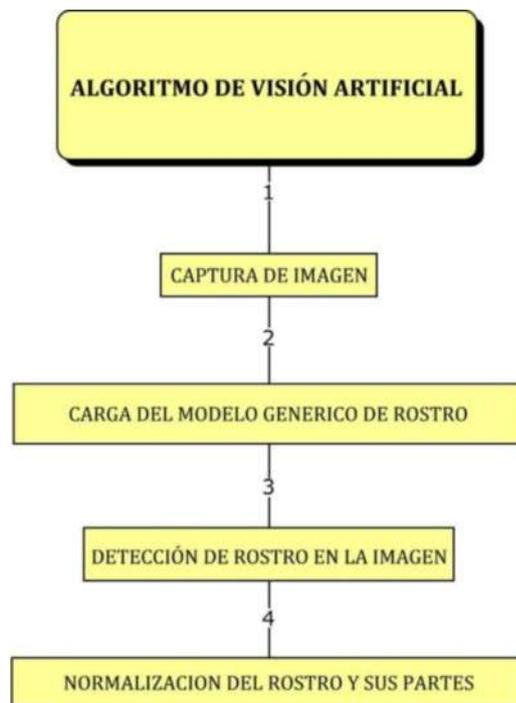
**Figura 26-2:** Código que permite decidir si es usuario registrado o no  
Realizado por: JARA, Caroline; CABA, Carlos 2018.

Como se ve en la figura 26-2 como primer paso se tiene almacenar el conjunto de resultados tanto de rostro, nariz, boca, ojos de cada usuario en una nueva variable para poder hacer modificaciones sin alterar el resultado. Luego se encuentra cual es el ganador de cada categoría y lo almacenamos. Cuando se tiene cada ganador se suma todo y se compara. Si existe un ganador claro se puede concluir que se ha detectado un usuario registrado y en cambio sí hay ganador por poco o igualdad se excluye la entrada y se deniega el acceso.

Como aclaración, se tiene en total ocho campos o categorías a analizar que son: rostro, nariz, ojos y boca tanto de frente como de perfil, y entonces depende de la precisión que deseemos para nuestro sistema podríamos aceptar como un ganador absoluto aquel que gane en cuatro de las categorías o más y desechar cualquier otro tipo de resultados.

## 2.5 Algoritmo de visión artificial

Uno de los dos puntos claves del proyecto es la utilización de un algoritmo de visión artificial, ya que para cualquier trabajo de tipo reconocimiento no solo facial, sino todo tipo de reconocimiento la visión artificial está presente. A continuación en la figura 27-2 se describe el algoritmo que hemos implementado y que nos ha dado los resultados apropiados para el proyecto.



**Figura 27-2:** Algoritmo de visión artificial  
Realizado por: JARA, Caroline; CABA, Carlos 2018

### 2.5.1 Captura de imagen

El primer paso para cualquier algoritmo de visión artificial siempre será la captura de la imagen, es la parte inicial y como su título indica no es más que capturar y guardar la imagen para poder procesarla luego. En el proyecto el código que permite realizar este paso del algoritmo es el siguiente:

```

vid2 = videoinput('winvideo', 2, 'MJPG_800x600');
set(vid2, 'TriggerRepeat', Inf);
vidRes2 = get(vid2, 'VideoResolution');
imWidth2 = vidRes2(1);
imHeight2 = vidRes2(2);
nBands2 = get(vid2, 'NumberOfBands');
hImage2 = image( zeros(imHeight2, imWidth2, nBands2), 'Parent', handles.axesFrente);
preview(vid2, hImage2);

dfotofrente=streat('fotos/', nombre, 'ff.jpg');
dfotoperfil=streat('fotos/', nombre, 'fp.jpg');

fotodefrente=getsnapshot(vid2);

imwrite(fotodefrente, dfotofrente);
imgf=imread(dfotofrente);

axes(handles.axesFrente);
imshow(imgf);

```

**Figura 28-2:** Código de captura de imagen en Matlab  
**Realizado por:** JARA, Caroline; CABA, Carlos 2018

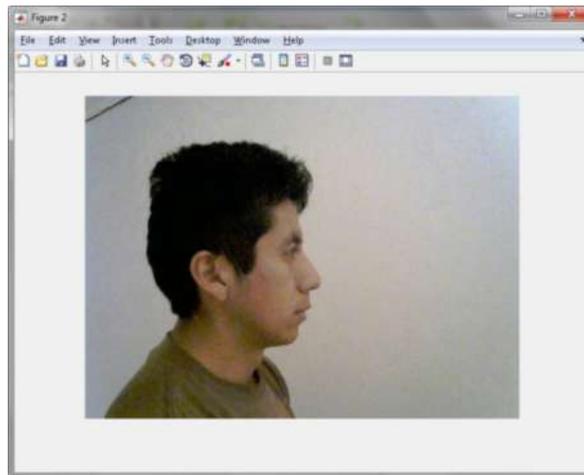
Como se puede ver en la figura 28-2 se inicia definiendo una variable que contiene el tipo de controlador que el computador tiene instalado para las cámaras, junto al número de dispositivo y la resolución con la que se va a trabajar. A continuación se realiza una pequeña configuración para poder mostrar en un axes de la guide un preview de la cámara. Luego se define la ruta en la cual se guardaran las imágenes luego de ser capturadas. Por último se ejecuta al fin la sentencia para capturar la foto y escribirla en la ruta previamente establecida.

Hay que aclarar que el código mostrado es únicamente de una de las cámaras usadas, como son idénticas las configuraciones para la otra es similar.

A continuación en la figura 29-2 y 30-2 se muestra la forma en la que el sistema obtiene el par de imágenes que serán procesadas durante el análisis.



**Figura 29-2:** Imagen de entrada de frente del usuario  
**Realizado por:** JARA, Caroline; CABA, Carlos 2018



**Figura 30-2:** Imagen de entrada de perfil del usuario  
Realizado por: JARA, Caroline; CABA, Carlos 2018

### 2.5.2 Carga del modelo genérico de rostro

Como el programa de desarrollo que se usa es Matlab este nos brinda la facilidad de almacenar y cargar conjunto de datos, más específicamente matrices de números. Este proceso de tener previamente guardado ese conjunto de datos y cargarlos es utilizado cuando es un conjunto extenso de datos y no se desea tener ningún tipo de pérdida.

A continuación se muestra el código que se utiliza para cargar el modelo del rostro con el que decidimos trabajar y también se muestra como se configuran sus parámetros.

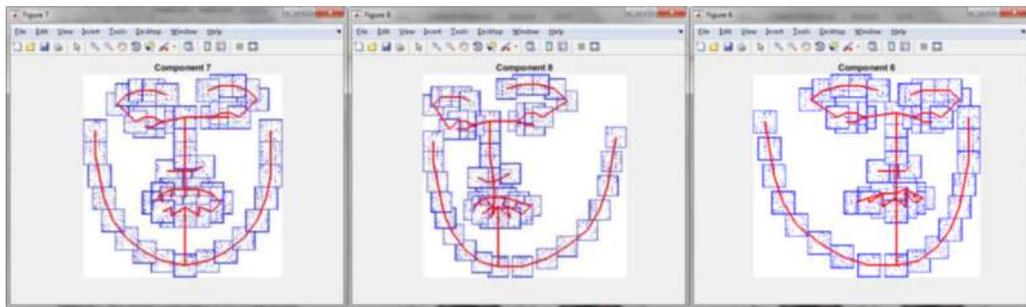
```
load face_p146_small.mat
% 5 levels for each octave
model.interval = 5;
% set up the threshold
model.thresh = min(-0.65, model.thresh);

if length(model.components)==13
    posemap = 90:-15:-90;
elseif length(model.components)==18
    posemap = [90:-15:15 0 0 0 0 0 0 -15:-15:-90];
else
    error('Can not recognize this model');
end
```

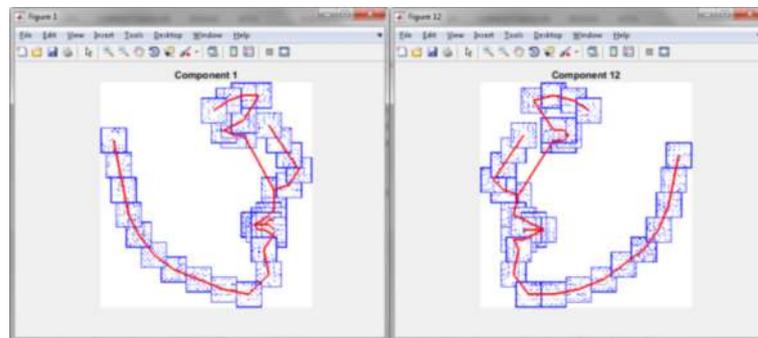
**Figura 31-2:** Código de carga del modelo en Matlab  
Realizado por: JARA, Caroline; CABA, Carlos 2018

Como se observa en la figura 31-2 se carga el modelo de rostro previamente almacenado con el nombre de face\_p146\_small.mat, luego se procede a configurar de acuerdo a las necesidades requeridas para ello se estableció su intervalo y sus límites. Luego se muestra un pequeño control en caso de que algo sucediera en la carga del modelo.

A continuación se muestra en las figuras 32-2 y 33-2, el modelo que se aplica para el proyecto tanto para la postura de frente como la de perfil.



**Figura 32-2:** Modelos *empleados* para el reconocimiento de frente  
Realizado por: JARA, Caroline; CABA, Carlos 2018



**Figura 33-2:** Modelos *empleados* para el reconocimiento de perfil  
Realizado por: JARA, Caroline; CABA, Carlos 2018

### 2.5.3 *Detección del rostro en la imagen*

La detección del rostro en una imagen su puede realizar de distintas formas, inclusive en Matlab existen desarrolladas varias funciones pero lo importante es saber cuál de ellas se elige y por eso mostramos a continuación el código del método que elegimos y explicamos su funcionamiento.

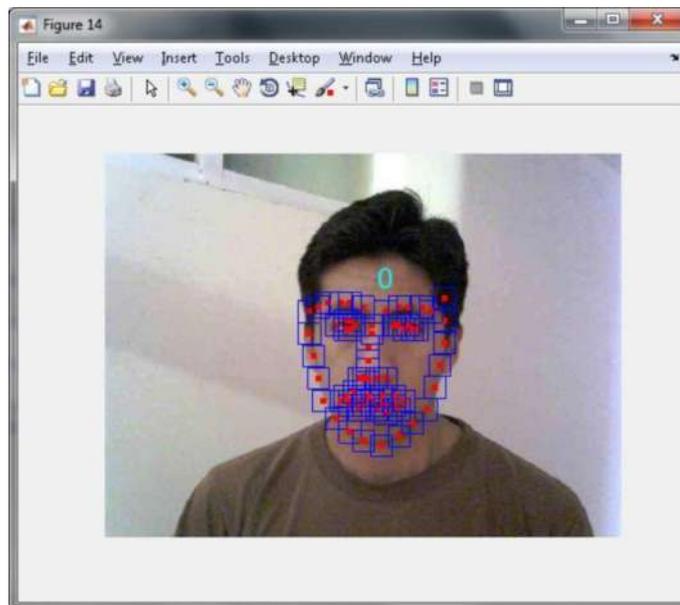
```
bs = detect(img, model, model.thresh);  
bs = clipboxes(img, bs);  
bs = nms_face(bs, 0.3);
```

**Figura 34-2:** Código de la detección de rostro en Matlab  
Realizado por: JARA, Caroline; CABA, Carlos 2018

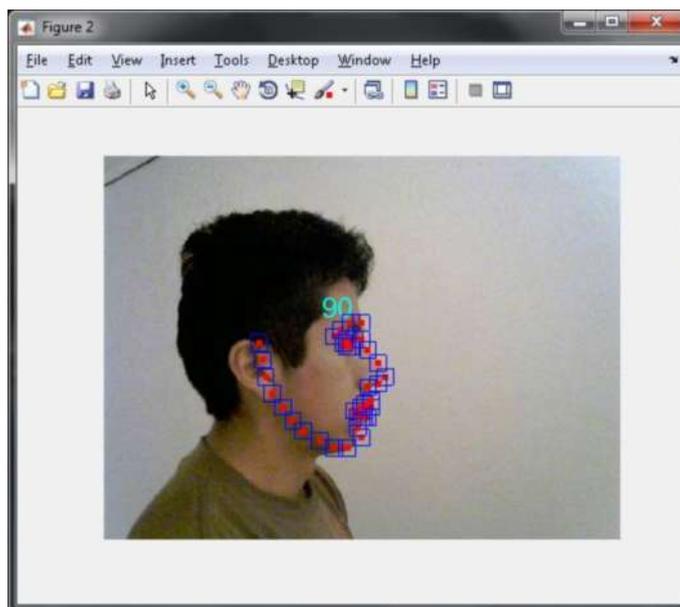
Como se explicó anteriormente ya existen herramientas desarrolladas y de código libre en Matlab para la detección de rostros, pero se trabajó con las tres funciones cuyo nombre están en la figura 34-2, cada función viene con su propia descripción de su autor y es la siguiente:

- La función detect viene descrita por el autor como función la cual consiste en buscar coincidencias o similitudes con poco margen de error entre la imagen de entrada y el modelo que se ha cargado previamente y nos da como respuesta un conjunto de puntos los cuales contiene dichas porciones de imagen en las que se ha encontrado algún similitud.
- La función clipboxes ahora se encarga de tomar cada una de las porciones de imagen que se obtuvieron en la función anterior y luego aplica un método de detección de bordes para poder ahora establecer si hay o no coincidencia en los rasgos del rostro cargado y no es un falso detectado.
- Por último la función nms\_face toma los puntos únicamente correspondientes a la cara detectada y los almacena como una matriz, esta función nos da como resultado el rostro detectado en base al modelo utilizado.

A continuación en la figuras 35-2 y 36-2 se muestra como el modelo nos permite realizar la detección del rostro en la imagen de entrada.



**Figura 35-2:** Detección del rostro en la imagen de frente  
 Realizado por: JARA, Caroline; CABA, Carlos 2018



**Figura 36-2:** Detección del rostro en la imagen de perfil  
 Realizado por: JARA, Caroline; CABA, Carlos 2018

#### 2.5.4 Normalización del rostro y sus partes

Por último paso de este algoritmo de visión artificial se procede a normalizar y obtener imágenes individuales de cada rostro y cada parte principal del mismo como son los ojos, la nariz y la boca. Para esta etapa se creó un código que se describe a continuación.

```

auxcortex=(bs.xy(:,1)+bs.xy(:,3))./2;
auxcortey=(bs.xy(:,2)+bs.xy(:,4))./2;
rostro=imcrop(img,[min(auxcortex) min(auxcortey) max(auxcortex)-min(auxcortex) max(auxcortey)-min(auxcortey)]);
rostro=imresize(rostro,[200 200]);

auxnarizx=(bs.xy(1:9,1)+bs.xy(1:9,3))./2;
auxnarizy=(bs.xy(1:9,2)+bs.xy(1:9,4))./2;
nariz=imcrop(img,[min(auxnarizx) min(auxnarizy) max(auxnarizx)-min(auxnarizx) max(auxnarizy)-min(auxnarizy)]);
nariz=imresize(nariz,[70 30]);

auxojosx=(bs.xy(10:31,1)+bs.xy(10:31,3))./2;
auxojosy=(bs.xy(10:31,2)+bs.xy(10:31,4))./2;
ojos=imcrop(img,[min(auxojosx) min(auxojosy) max(auxojosx)-min(auxojosx) max(auxojosy)-min(auxojosy)+20]);
ojos=imresize(ojos,[30 170]);

auxbocax=(bs.xy(32:51,1)+bs.xy(32:51,3))./2;
auxbocay=(bs.xy(32:51,2)+bs.xy(32:51,4))./2;
boca=imcrop(img,[min(auxbocax) min(auxbocay) max(auxbocax)-min(auxbocax) max(auxbocay)-min(auxbocay)]);
boca=imresize(boca,[30 70]);

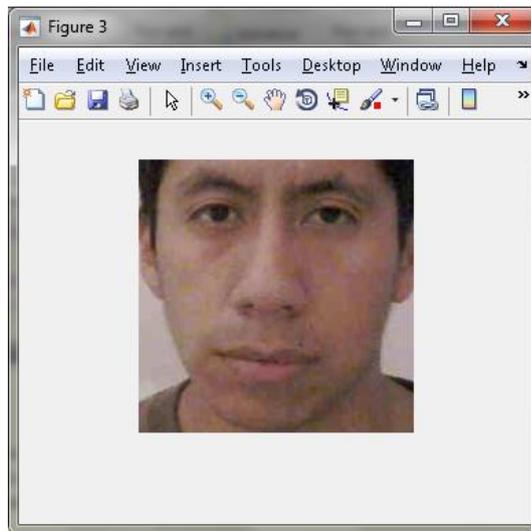
```

**Figura 37-2:** Código de normalización y recorte de rostro y sus partes  
 Realizado por: JARA, Caroline; CABA, Carlos 2018.

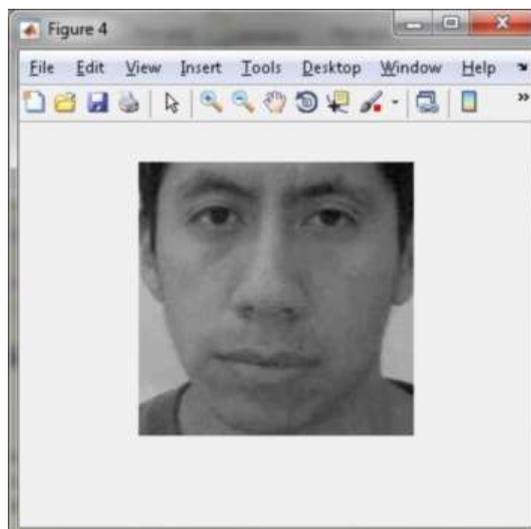
Como se puede observar en la figura 37-2 hay cuatro secciones similares que contienen en primer lugar una selección de los puntos obtenidos anteriormente, esta selección dependerá de que parte del rostro está siendo procesada por ejemplo para el rostro entero se utilizan todos los puntos, en cambio para la nariz se utilizan los nueve primeros puntos y así como se observa.

Luego de esta selección se hace el recorte de la imagen y por último punto la normalización es decir definir el tamaño con el cual se va a trabajar.

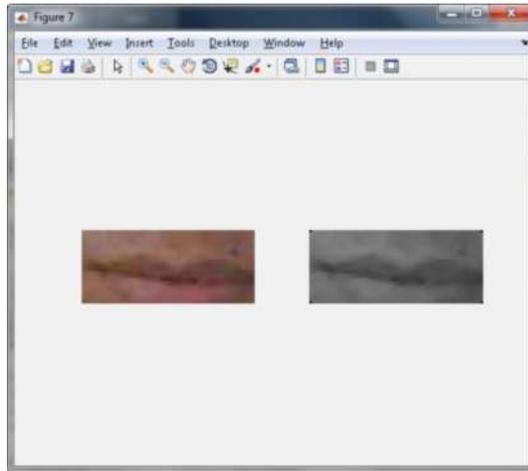
Para la captura de imagen de frente y de perfil es la misma lógica de programación, pero lo único que cambia son la cantidad de puntos encontrados y seleccionados. A continuación en las figuras 38-2 hasta la figura 47-2 vemos de forma clara la normalización de las imágenes.



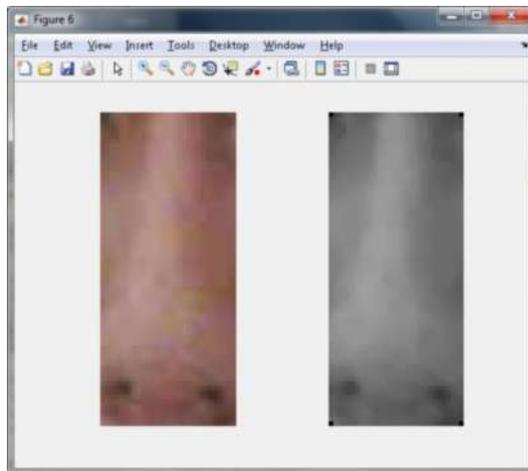
**Figura 38-2:** Recorte del rostro de frente  
Realizado por: JARA, Caroline; CABA, Carlos 2018



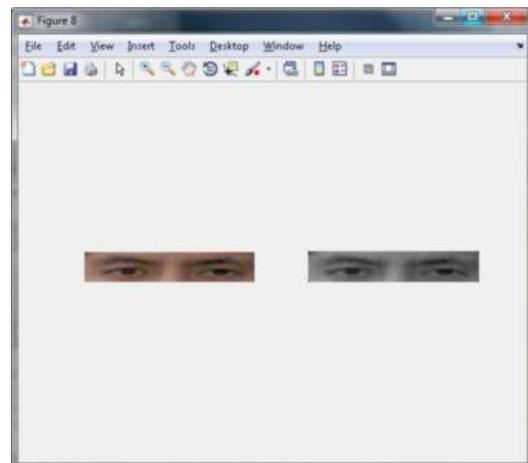
**Figura 39-2:** Normalización del rostro de frente  
Realizado por: JARA, Caroline; CABA, Carlos 2018



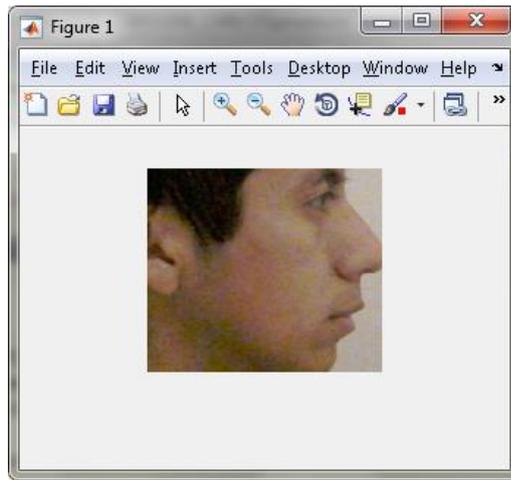
**Figura 40-2:** Recorte y normalización boca  
Realizado por: JARA, Caroline; CABA, Carlos 2018



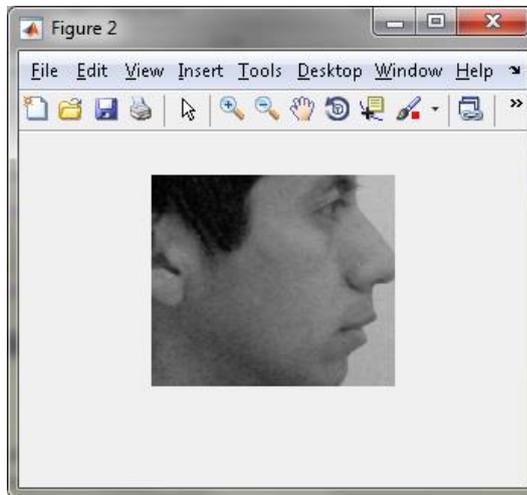
**Figura 41-2:** Recorte y normalización nariz  
Realizado por: JARA, Caroline; CABA, Carlos 2018



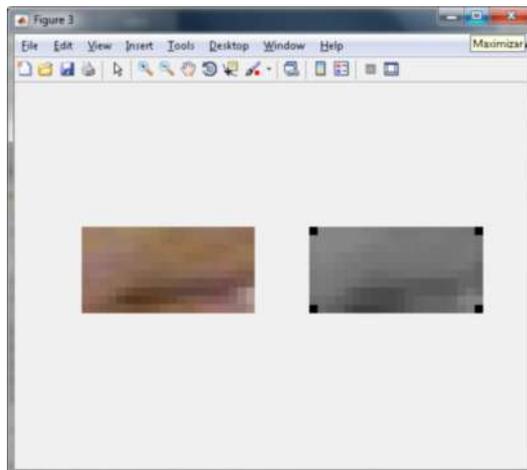
**Figura 42-2:** Recorte y normalización de ojos  
Realizado por: JARA, Caroline; CABA, Carlos 2018



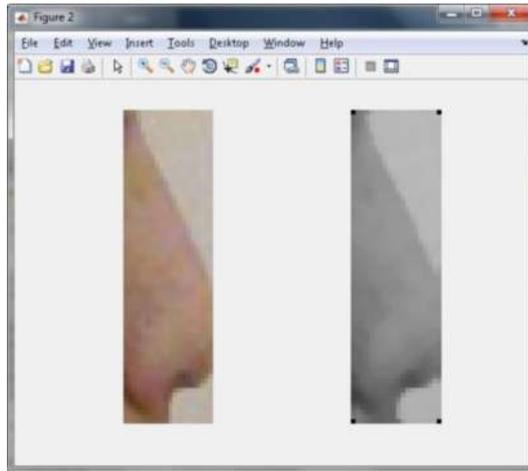
**Figura 43-2:** Recorte del rostro de perfil  
Realizado por: JARA, Caroline; CABA, Carlos 2018



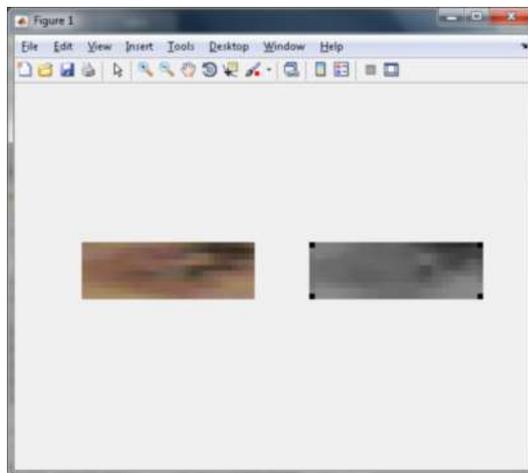
**Figura 44-2:** Normalización del rostro de perfil  
Realizado por: JARA, Caroline; CABA, Carlos 2018



**Figura 45-2:** Recorte y normalización boca  
Realizado por: JARA, Caroline; CABA, Carlos 2018



**Figura 46-2:** Recorte y normalización nariz  
Realizado por: JARA, Caroline; CABA, Carlos 2018

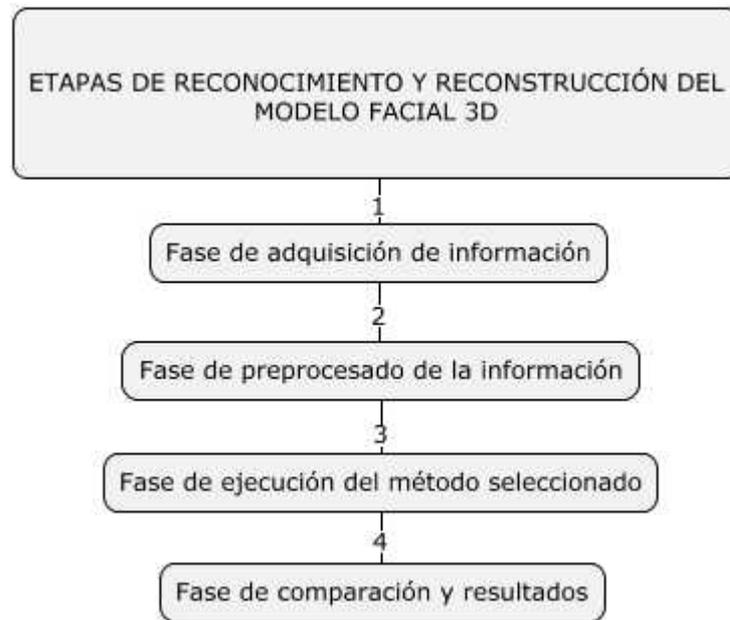


**Figura 47-2:** Recorte y normalización de ojos  
Realizado por: JARA, Caroline; CABA, Carlos 2018

## **2.6 Descripción de las etapas del reconocimiento y reconstrucción 3D**

En esta sección se describirá de una forma comprensible y detallada los procesos del funcionamiento e implementación del proyecto, así como los distintos puntos clave que se presentan en el sistema elaborado.

Para una mejor comprensión del funcionamiento del sistema se presenta un esquema como se indica en la figura 48-2 con las etapas que lo conforman y asimismo se describen más adelante cada una de ellas.



**Figura 48-2:** Etapas del reconocimiento y creación del modelo facial 3D  
 Realizado por: JARA, Caroline; CABA, Carlos 2018.

Cabe mencionar que el proyecto está programado en el software Matlab, y para facilidad de los autores se hizo uso de herramientas desarrolladas en el mismo, de código libre para usos puntuales pero que son de gran aporte para el resultado final esperado.

Adicional se menciona que la elaboración de la parte lógica (programa) está estrictamente acondicionada para este proyecto y para los resultados planteados anteriormente, esto quiere decir que nuestro modelo podrá ser únicamente aplicado a la seguridad de ingreso a un determinado lugar por medio de sistema de video.

En cuanto a la parte física del sistema, en su totalidad se decidió implementarlo de forma interna en uno de los hogares de los autores, simulando así cualquier tipo de ingreso a determinado lugar en condiciones arbitrarias. También se ha diseñado una interface gráfica de monitoreo en tiempo real para así abarcar todos los campos correspondientes a la seguridad.

### **2.6.1 Fase de adquisición de información**

#### *2.6.1.1 Descripción.*

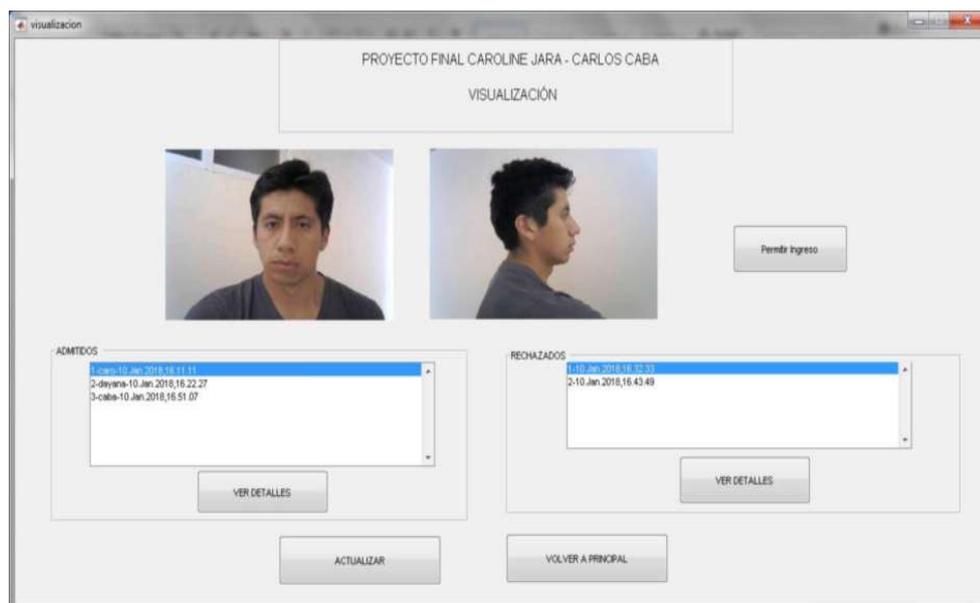
Esta etapa abarca todo lo relacionado a la captura de imágenes por medio de las cámaras, para el caso en mención las imágenes las tomamos con un par de Webcam de marca Logitech c270 a

una resolución de imagen fijada en 800x600 pixeles; el proceso siendo más específicos funciona de la siguiente manera:

- Si fuera el caso de un nuevo usuario que se está registrando en la base de datos de nuestro sistema, la captura de fotografías lo debe realizar una segunda persona asegurándose que la posición de la persona y demás características asociadas a la imagen estén dentro de lo permitido.
- Por otra parte si fuera un usuario que está solicitando acceso la adquisición de las fotografías está relacionado con un botón pulsador que está programado para permitir en el instante la captura de las fotografías. Cabe recalcar que en este caso no hay ningún veedor que verifique el correcto posicionamiento de la persona, simplemente el usuario deberá esperar la concesión o no del permiso de ingreso que el sistema luego de realizar todo el proceso le conceda.

Sin importar cual fuese el caso dentro de esta etapa también se ejecutan las sentencias para almacenar las imágenes en la base de datos, para nuestro proyecto se creó una base de datos en el software MySQL Workbench 6.3 CE y se enlazo a nuestro entorno de desarrollo Matlab R2015a.

A continuación en la figura 49-2 se ve unas imágenes en donde se muestran las fases de adquisición de la información.



**Figura 49-2:** Adquisición de imágenes del usuario que solicita acceso  
**Realizado por:** JARA, Caroline; CABA, Carlos 2018.

### 2.6.1.2 Comandos y funciones de Matlab

**Tabla 1-2:** Comandos principales para la adquisición de información

COMANDO PRINCIPALES	DESCRIPCIÓN
videoinput();	Permite configurar la controladora, el número y la resolución de la cámara a utilizar.
preview () ;	Permite iniciar un visualizador en pantalla de lo captado por la cámara.
stoppreview();	Permite detener el visualizador de imagen.
getsnapshot();	Permite guardar en una variable temporal la imagen que visualiza la cámara.
image();	Trata los datos de una matriz como una imagen.
imwrite();	Permite guardar una imagen en un espacio de memoria del computador.
imread();	Permite leer y copiar en una variable temporal una imagen del computador.
database();	Permite crear una variable de conexión entre Matlab y MySql.
exec();	Permite ejecutar una sentencia query dentro del entorno de Matlab.

Fuente: (Mathworks, 1994-2018)

**Tabla 2-2:** Comandos auxiliares para la adquisición de información

COMANDOS AUXILIARES	DESCRIPCIÓN
strcat();	Permite concatenar varias cadenas en una sola.
get();	Permite obtener variables, características o valores del entorno GUIDE de Matlab.
set();	Permite establecer variables, características o valores del entorno GUIDE de Matlab.
imshow();	Muestra en pantalla una imagen.

Fuente: (Mathworks, 1994-2018)

## 2.6.2 Fase de preprocesado de la información

### 2.6.2.1 Descripción

Una vez que ya contamos con las imágenes almacenadas en la base de datos el siguiente punto es su pre procesamiento y esto abarca lo siguiente:

- **Recorte de rostro:** Consiste en usar nuestra función denominada *procesarimagen()*; y lo que realiza es usar un modelo genérico de rostro para cada posición de entrada (imagen de frente y de perfil), con lo cual procede a buscar y recorta el rostro de la persona.
- **Recorte de partes del rostro:** Usando la misma ideología del caso anterior se comparan los ojos, nariz y boca del modelo genérico que usamos y se recorta cada imagen.
- **Reajuste de tamaño:** Luego de tener cada imagen identificada correctamente se procede a dar un tamaño preestablecido a cada una, en nuestro sistema se consideró aceptable trabajar con los tamaños como se indican en la tabla 3-2:

**Tabla 3-2:** Tamaños de imágenes normalizadas

IMAGEN	TAMAÑO (píxeles)
Rostro de frente	200x200
Ojos de frente	30x170
Nariz de frente	70x30
Boca de frente	30x70
Rostro de perfil	140x160
Ojos de perfil	10x30
Nariz de perfil	70x20
Boca de perfil	10x20

Realizado por: JARA, Caroline; CABA, Carlos 2018.

- **Almacenamiento de cambios:** Una vez que hemos identificado y recortado cada una de las partes del rostro que necesitamos procedemos a almacenarlas en la base de datos con el fin de tener más precisión al trabajar con el reconocimiento usando más datos, cada una de las nuevas imágenes obtenidas se asociará con el id del usuario correspondiente.

Para el sistema que elaboramos se consideró apropiado usar una tabla denominada *usuarios* que se compone de la siguiente forma como se muestra en la tabla 4-2.

**Tabla 4-2:** Tabla de usuarios creada en la base de datos

<b>CAMPO</b>	<b>TIPO DE DATO – TAMAÑO</b>	<b>FINALIDAD EN NUESTRO PROYECTO</b>
ID	Int – 10	Almacena la id para una fácil manipulación de usuarios.
Nombre	Varchar - 45	Almacena nombre de los usuarios.
Clave	Varchar - 45	Contiene la cadena de caracteres que el usuario haya designado como contraseña.
Rostrof	Varchar – 45	Contiene la dirección de la foto del rostro de frente.
Narizf	Varchar - 45	Contiene la dirección de la foto de la nariz de frente.
Bocaf	Varchar – 45	Contiene la dirección de la foto de la boca de frente.
Ojosf	Varchar - 45	Contiene la dirección de la foto de los ojos de frente.
Rostrop	Varchar – 45	Contiene la dirección de la foto del rostro de perfil.
Narizp	Varchar – 45	Contiene la dirección de la foto de la nariz de perfil.
Bocap	Varchar – 45	Contiene la dirección de la foto de la boca de perfil.
Ojosp	Varchar – 45	Contiene la dirección de la foto de los ojos de perfil.

Realizado por: JARA, Caroline; CABA, Carlos 2018

#### 2.6.2.2 Comandos y Funciones de Matlab

**Tabla 5-2:** Comandos principales para la fase de preprocesado de la imagen

<b>COMANDOS PRINCIPALES</b>	<b>DESCRIPCIÓN</b>
load	Permite cargar el modelo general del rostro que vayamos a utilizar.
imcrop	Permite recortar una imagen dados los puntos del rectángulo que la contiene.

imresize	Permite cambiar el tamaño de una imagen a cualquier tamaño especificado sin recortarla
----------	----------------------------------------------------------------------------------------

Fuente: (Mathworks, 1994-2018)

**Tabla 6-2:** Funciones para la fase de preprocesado de la imagen

FUNCIONES	DESCRIPCIÓN
procesarimagen	Elaborada para tomar como entrada una imagen en donde se pueda observar un rostro (ya sea de frente o de perfil) y la compara con un modelo genérico de rostro y obtiene un contorno sobrepuesto de lo cual procede a cortar el rostro y cada parte que lo conforma (ojos, nariz, boca) y los devuelve como imágenes individuales

Realizado por: JARA, Caroline; CABA, Carlos 2018.

### 2.6.3 Fase de ejecución del método seleccionado

#### 2.6.3.1 Descripción

Teniendo las imágenes de cada rostro y cada parte del rostro de los usuarios ahora el siguiente paso sería realizar una comparación, dicha comparación debe ser especialmente con técnicas para procesar imágenes ya que realizar comparación bit a bit de imágenes sería desperdicio de tiempo, recursos y dinero. El motivo por el que no es posible una comparación bit a bit es debido a que una pequeña variación en uno de los parámetros no controlados del ambiente (cantidad de luz, reflexión de luz en el ambiente, absorción de luz, etc) así como también de las características del usuario que usa nuestro sistema (color de piel, rasgos faciales, etc) representa un gran cambio casi imperceptible al ojo humano pero que afecta mucho a nivel de matrices.

El método que hemos adoptado para realizar el reconocimiento facial consiste aplicar filtros (wavelets) de Gabor los cuales entre otras funciones son usados para comparación entre imágenes.

El proceso de comparación consiste de varias etapas:

- **Dividir en celdas la imagen de entrada:** Consiste en tomar la imagen de entrada y dependiendo el tamaño que tenga lo dividirá en celdas iguales con el fin de minimizar el espacio en el cual vamos a aplicar los filtros de Gabor para así tener un mayor porcentaje de similitud. Cabe mencionar que al tener ya valores de tamaños fijos para cada imagen este proceso dividirá en igual número de partes a cada imagen y así se evitarían posibles errores en la comparación.

- **Aplicar filtros (wavelets) de Gabor a cada imagen:** Usando nuestra función *obtenervectorescaracteristicos()*, se realiza el proceso de dividir cada imagen en celdas que se mencionó anteriormente y adicional se procede a usar el filtro en cada sección de imagen y así luego tener un conjunto de vectores que serán los cuales nos sirvan para la fase de identificación de usuario.
- **Guardar cada banco de vectores:** Cuando nos referimos a bancos de vectores estamos hablando del conjunto de valores que obtuvimos como resultado de aplicar los filtros de Gabor a cada imagen de entrada; estos vectores variaran en función de cada rasgo facial ya que al ser un conjunto de filtros rotacionales identifican casi cualquier tipo de rasgo. Una vez que tengamos dichos vectores el siguiente paso es guardarlos temporalmente para un proceso posterior en el cual se tomará una acción correspondiente.

#### 2.6.3.2 Comandos y Funciones de Matlab

**Tabla 7-2:** Comandos principales para la fase de ejecución del método seleccionado

COMANDOS PRINCIPALES	DESCRIPCIÓN
zeros()	Permite crear una matriz de zeros de las dimensiones que desee el programador
fix()	Permite redondear un número decimal al entero más próximo a cero que exista.

Fuente: (Mathworks, 1994-2018)

**Tabla 8-2:** Funciones para la fase de ejecución del método seleccionado

FUNCIONES	DESCRIPCIÓN
ajustarimagen	Esta función cumple con la primera etapa de esta sección que es dividir en partes iguales la imagen de entrada.
G	Esta función la desarrollamos únicamente para que aplique la función Gabor (filtro de Gabor) a cada imagen previamente dividida y nos del resultado en forma de banco de vectores.
obtenervectorescaracteristicos	Función diseñada para mandar a llamar a las funciones antes descritas en el momento indicado y también realizar el acople

Realizado por: JARA, Caroline; CABA, Carlos 2018.

## 2.6.4 Fase de comparación y resultados

### 2.6.4.1 Descripción

Como en el paso previo ya se obtuvo un banco de vectores de cada imagen el siguiente paso es compararlo con el banco de vectores que produzcan las imágenes almacenadas en la base de datos correspondiente a los usuarios registrados y por ende que posean acceso, para esto se realizará una comparación usando el método de distancia de Euler que es fácilmente aplicable aquí ya que no es más que almacenar todos los bancos de la base en un mismo vector y luego compararlos con el banco de la imagen de entrada y así verificar si hay un ganador o no, es decir si el usuario que pide acceso está o no registrado.

Para explicar de mejor manera esta etapa del sistema a continuación mostramos cada paso y su descripción:

- **Agrupar toda la información y comparar:** Consiste en obtener todos los bancos de vectores de las imágenes de cada persona en la base de datos y crear una sola matriz con todos estos para luego realizar la comparación. Para aclarar se realiza procesos separados para cada tipo de imagen; es decir, un proceso para las imágenes del rostro, otro proceso para los ojos, otro para la nariz, etc. tanto de frente como de perfil.
- Verificar si existe algún ganador: Como luego de comparar cada imagen de cada usuario tenemos un valor de similitud, para verificar si hay o no un ganador se procede a usar uno de los métodos de la inteligencia artificial denominado lógica difusa que consiste en dar un rango de valores para cada situación posible y luego tomar una decisión, en este caso se procede a verificar si se ha hallado todas las imágenes posible, si es así se ejecuta la comparación individual de cada parte para luego un conteo de valores en los que el usuario salió ganador y así tomar una decisión.
- Similar proceso se toma en caso de no detectar alguna de las partes del usuario pero variando los porcentajes proporcionados a las demás partes y también variando el rango permitido de falsos detectados.

- Almacenamiento de resultados: Sea cualquiera el resultado de la comparación del usuario se debe almacenar, para eso se cuenta con dos tablas en la base de datos que contiene lo siguiente indicado en las tablas 9-2 y 10-2:

**Tabla 9-2:** Tabla de admitidos creada en la base de datos

<b>CAMPO</b>	<b>TIPO DE DATO – TAMAÑO</b>	<b>FINALIDAD EN NUESTRO PROYECTO</b>
ID	Int - 10	Almacena la id para una fácil manipulación de usuarios.
Nombre	Varchar - 45	Almacena nombre de los usuarios.
Rostrof	Varchar – 45	Contiene la dirección de la foto del rostro de frente.
Rostrop	Varchar – 45	Contiene la dirección del rostro de perfil.
Similitud	Varchar - 45	Contiene el porcentaje de similitud con el cual se consideró admitido el usuario.
horayfecha	Varchar - 45	Contiene la hora y fecha en el que se ha considerado como admitido el usuario.

Realizado por: JARA, Caroline; CABA, Carlos 2018.

**Tabla 10-2:** Tabla de rechazados creada en la base de datos

<b>CAMPO</b>	<b>TIPO DE DATO – TAMAÑO</b>	<b>FINALIDAD EN NUESTRO PROYECTO</b>
ID	Int - 10	Almacena la id para una fácil manipulación de usuarios.
rostrof	Varchar – 45	Contiene la dirección de la foto del rostro de frente de la persona rechazada.
horayfecha	Varchar - 45	Contiene la hora y fecha en el que se a considerado como

		rechazado el usuario.
--	--	-----------------------

Realizado por: JARA, Caroline; CABA, Carlos 2018.

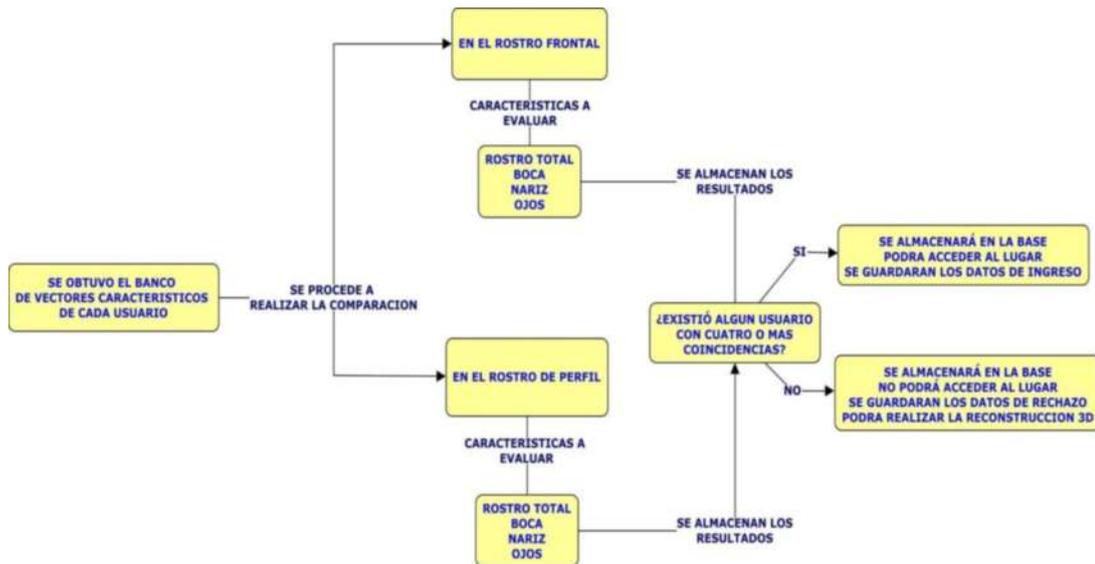
- **Ejecución de determinada acción:** Una vez que ya se toma una decisión ahora el siguiente punto es que debe hacer el programa, para lo cual como ya se ha mencionado antes existen dos posibles resultados que la persona esté o no registrada en la base de datos.

Para el caso que la persona esté registrada en la base de datos se procederá a ejecutar la conexión entre Matlab y nuestro controlador que será el Arduino, luego se activa una salida del mismo que conectada a la placa que diseñamos y que más adelante se muestra su composición activará un circuito que da paso a accionar la chapa eléctrica que se ha colocado en la puerta de ingreso, a la vez que reproducirá un saludo programado para el usuario.

Ahora bien, en el caso que la persona que ha solicitado el acceso no esté registrada el sistema aparte de almacenar su información no ejecutará ninguna otra acción y volverá al estado de espera por posibles nuevos usuarios.

Otra opción que muestra el sistema es la posibilidad de evidenciar la actividad que se ha suscitado, es decir podemos ver que permisos de acceso se ha producido y a qué hora y fecha. Idéntica forma tenemos de ver los usuarios rechazados, su fecha y hora de solicitud de ingreso, y un punto adicional especialmente para estos últimos se realizará la reconstrucción 3D en el caso que el administrador del sistema así lo decida.

Para una mejor comprensión de lo que se refiere a la toma de decisión del usuario que solicitó ingreso al sistema a continuación se muestra en la figura 50-2 un pequeño diagrama de flujo en donde se explica que acciones se toman.



**Figura 50-2:** Acciones que se toman cuando un usuario es aceptado o no  
**Realizado por:** JARA, Caroline; CABA, Carlos 2018

A continuación en la figura 51-2 se muestra una de las opciones de resultado de la creación del modelo facial en 3D de un usuario, esta opción estará disponible únicamente para los usuarios rechazados.



**Figura 51-2:** Modelo facial 3D  
**Realizado por:** JARA, Caroline; CABA, Carlos 2018

## CAPÍTULO III

### 3 MARCO DE RESULTADOS, ANÁLISIS Y DISCUSIÓN DE RESULTADOS

#### 3.1 Análisis para determinar el método de detección facial (Objetivo 1)

##### 3.1.1 Análisis en base al tiempo de respuesta en distintas posiciones

Existen varios algoritmos para la detección de rostros. Cada uno de estos algoritmos está basado alguna técnica conocida de reconocimiento de formas, cada detector tiene características que lo hacen diferente al resto. El detector de Viola-Jones con el que se realiza la comparación es un mecanismo ampliamente utilizado para la detección de objetos porque funciona mejor en objetos frontales y no tan bien en las vistas laterales. La principal propiedad de este algoritmo es que el entrenamiento es lento, pero la detección es rápida, pero no es capaz de calcular la orientación de la cara ni darnos una estimación de lo seguro que está de la detección.

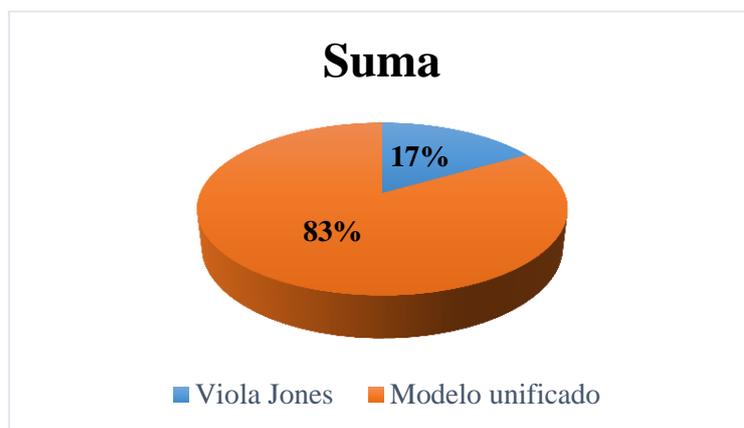
Se realiza un análisis de comparación entre dos algoritmos de detección Viola Jones y el modelo unificado basados en el tiempo que se demora en detectar el rostro de una persona en diferentes ángulos de posición. La siguiente tabla 1-3 muestra los datos tomados con los dos métodos en condiciones normales.

**Tabla 1-3:** Tiempo de respuesta de los dos métodos.

<b>Tiempo de respuesta en detección facial</b>		
	<b>Viola Jones</b>	<b>Modelo unificado</b>
DetECCIÓN a 90°	X	17,4452
DetECCIÓN a 70°	X	17,3805
DetECCIÓN a 50°	X	17,0265
DetECCIÓN a 30°	X	16,3671
DetECCIÓN a 10°	16,5867	15,3248
DetECCIÓN a -10°	16,7963	15,1729
DetECCIÓN a -30°	X	16,3147
DetECCIÓN a -50°	X	17,0393
DetECCIÓN a -70°	X	17,3761
DetECCIÓN a -90°	X	17,4659
<b>Suma</b>	33,383	166,913

Realizado por: JARA, Caroline; CABA, Carlos 2018.

Mediante la tabla 1-3 se escoge el método de detección del modelo unificado que es el de mayor suma como se puede ver en el gráfico 1-3 ya que el método de Viola Jones solo detecta en variaciones de posición de  $-10^{\circ}$  a  $10^{\circ}$  por eso su tiempo de respuesta es menor y al existir una mínima diferencia de tiempo entre ambos métodos no conviene utilizar el primer método ya que el sistema requiere dos fotos de ingreso una de frente y de perfil.



**Gráfico 1-3:** Porcentajes de los métodos en tiempo de respuesta  
Realizado por: JARA, Caroline; CABA, Carlos 2018

### 3.1.2 *Análisis en base al número de intentos bajo ciertas condiciones*

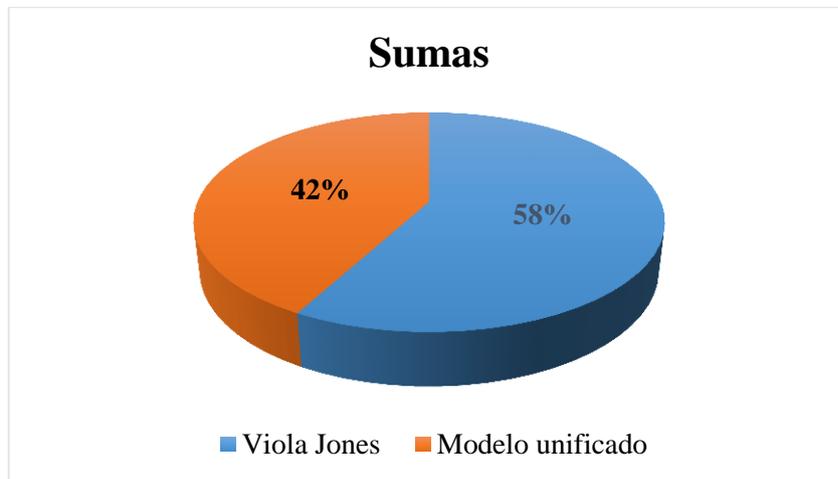
Se procedió un análisis de comparación entre dos métodos de detección basados en el número de intentos para detectar un rostro de acuerdo a ciertas condiciones que se describen en la tabla. La siguiente tabla 2-3 muestra los datos obtenidos con los dos métodos en las siguientes condiciones.

**Tabla 2-3:** Intentos de detección bajo ciertas condiciones

Número de intentos en detección facial bajo las siguientes condiciones		
	Viola Jones	Modelo unificado
Poca luz	3	2
Luz adecuada	1	1
Fondo uniforme	1	1
Variación de fondo	3	2
Poca distancia a la cámara	1	1
Variación distancia a la cámara	1	1
<b>Suma</b>	<b>10</b>	<b>8</b>

Realizado por: JARA, Caroline; CABA, Carlos 2018

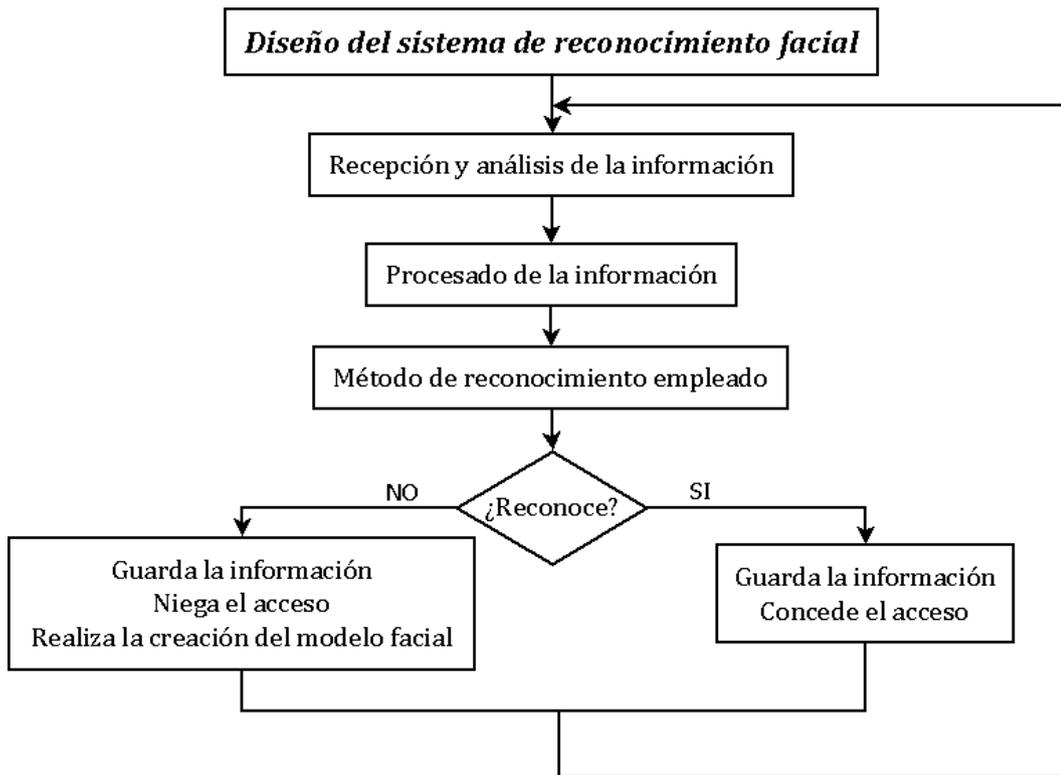
En base a esto concluimos que el método que se escoge es el del modelo unificado ya que requiere de menos condiciones en cuanto a detección y como se puede ver en la tabla 3-3 se escoge al de menor suma porque presenta menos intentos para detectar un rostro en esas condiciones esos valores se encuentran expresados en el gráfico 2-3.



**Gráfico 2-3:** Porcentajes de los métodos bajo ciertas condiciones  
Realizado por: JARA, Caroline; CABA, Carlos 2018

### 3.2 Diseño final del sistema (Objetivo 2 y 4)

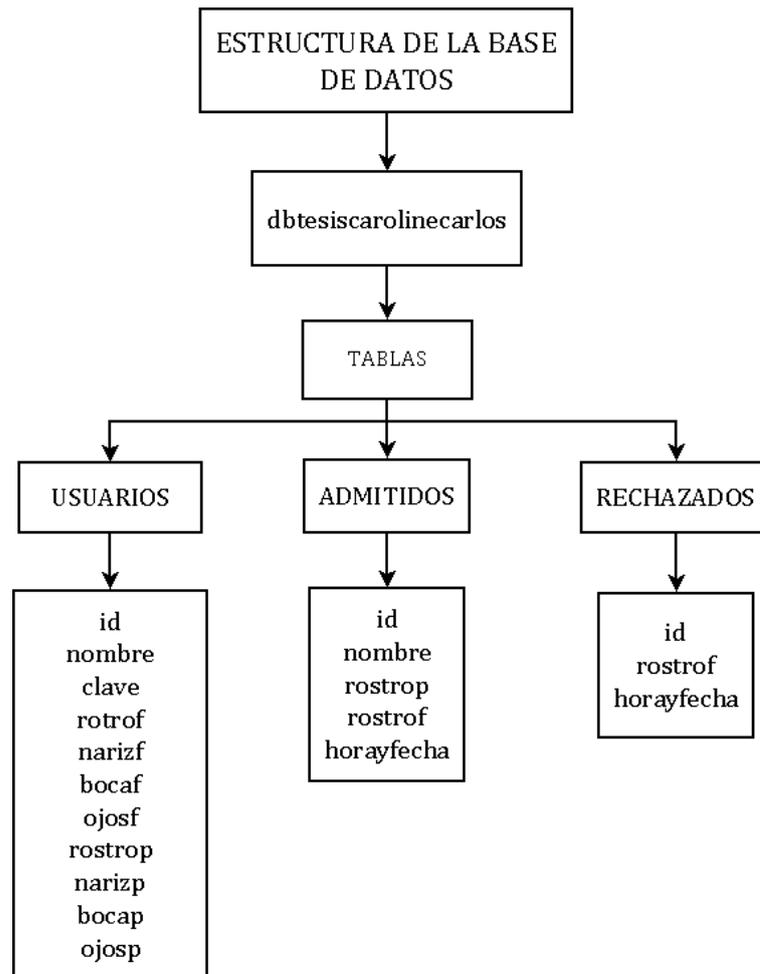
El diseño utilizado para el desarrollo del sistema inicia con la recepción y análisis de la información, aquí abarca todo lo relacionado a la captura de las imágenes; luego tenemos el procesado de la información, que hace referencia a detección facial y normalización de imágenes; una vez procesada la información de entrada realizamos la ejecución del método de reconocimiento que en nuestro caso es el método EBGM, luego tenemos dos etapas finales dependientes del resultado del reconocimiento; por un lado en caso de no reconocer se guardará la información en la base de datos, luego se negará el acceso al lugar y por último se realizará la creación del modelo facial 3D; por otra parte cuando el reconocimiento sea exitoso se almacenará la información del usuario en la base de datos y se concederá el acceso al lugar; una vez culminado todo el proceso se reinicia ante la existencia de nueva información de entrada. A continuación en la figura 1-3 se muestra el diseño del sistema descrito.



**Figura 1-3:** Diseño del sistema de reconocimiento facial  
 Realizado por: JARA, Caroline; CABA, Carlos 2018.

### 3.3 Estructura de la base de datos implementada (Objetivo 3).

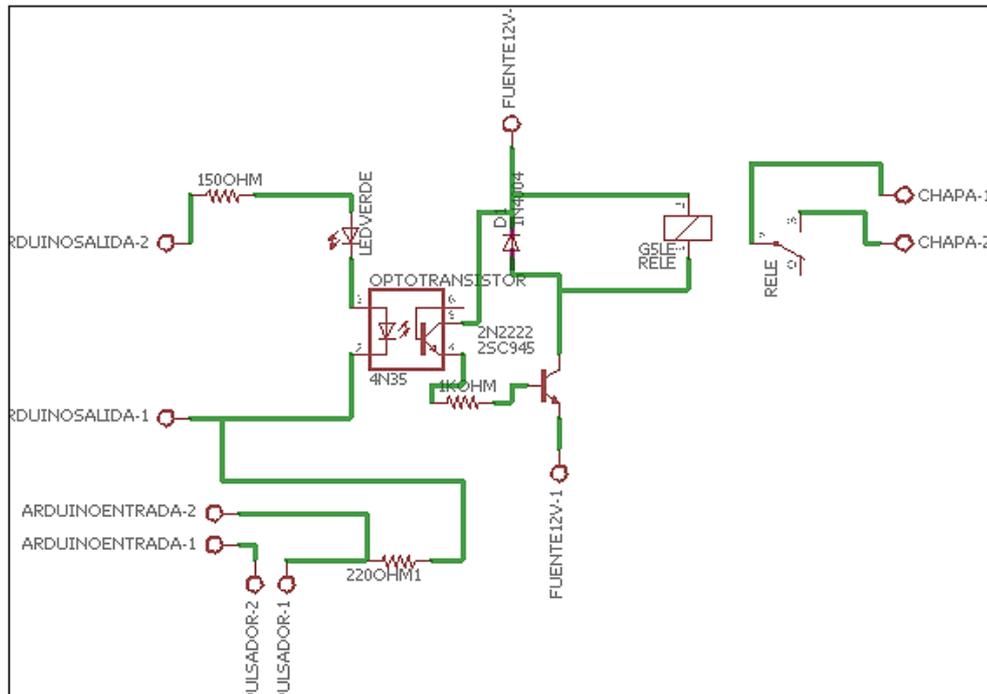
Para la ejecución del sistema se decide usar una base de datos estructurada como se indica en la figura 2-3 la cual cuenta con tres tablas; la tabla de usuarios está destinada a almacenar la información de aquellas personas que se han registrado en la base de datos, la información que contiene es su número id de identificación para la base de datos, el nombre y clave personal para cada usuario, y la dirección del directorio que contiene la información de las imágenes correspondientes a dicha persona; la tabla de admitidos contiene el nombre de usuario admitido y la información de la imagen de frente y de perfil de dicha persona, además contiene la hora y fecha en la cual se dio paso ha dicho usuario; por último la tabla de rechazados contiene únicamente la hora y fecha en la cual ha solicitado el acceso la persona no registrada y su imagen de frente la cual se usa para la creación del modelo facial en 3D.



**Figura 2-3:** Diseño del sistema de reconocimiento facial  
 Realizado por: JARA, Caroline; CABA, Carlos 2018.

### 3.4 Conexión entre el software y los actuadores (Objetivo 5)

Para el proyecto se realizó una placa capaz de interpretar los comandos del proyecto y transformarlos a órdenes que interpreten los dispositivos de salida (cerradura eléctrica), esta placa está conformada como se indica en la figura 3-3. Una de las funciones importantes de la placa aparte de interactuar entre los actuadores y el software, es la de proteger al controlador que es la computadora de un posible sobre voltaje por parte del circuito de potencia.



**Figura 3-3:** Circuito de control de actuadores  
**Realizado por:** JARA, Caroline; CABA, Carlos 2018.

Ahora bien el circuito se puede dividirlo en tres partes: La primera conformada por las borneras denominadas *ARDUINOSALIDA-1* y *ARDUINOSALIDA-2*, la resistencia de *150OHM*, el *LEDVERDE* y el opto transistor *4N35* tiene como finalidad aislar por completo la parte de mando que funciona con 12V DC de las entradas/salidas del controlador que trabajan con 5V DC y también acoplan su acción de tal manera que se pueda controlar la acción de la chapa eléctrica con una salida digital del Arduino.

La segunda parte del circuito se conforma por las borneras *ARDUINOENTRADA-1*, *ARDUINOENTRADA-2*, *PULSADOR-1*, *PULSADOR-2*, y la resistencia de *220OHM* tienen como finalidad detectar cuando el usuario solicita acceso al sistema y a su vez también proporcionan de alimentación de los 5V DC para el funcionamiento de la primera parte del circuito.

Por ultimo tenemos la denominada “parte de mando” que lo conforman la resistencia de *1KOHM*, el transistor *2N2222*, el diodo *1N4004*, el relé de 12V DC y las borneras denominadas *FUENTE12V-1*, *FUENTE12V-2*, *CHAPA-1*, *CHAPA-2* cumplen con la función de tomar una señal de entrada proveniente del arduino por medio del opto acoplador en la base del transistor que está configurado como interruptor y permitir el paso de corriente a través del relé lo cual activará sus contactos y logrará energizar la chapa eléctrica que admite como voltaje de

activación 12V DC, adicional por un par de sus borneras será donde se suministre la fuente de 12V DC a 3A.

La razón por la se decidió usar esa fuente es debido a que la chapa eléctrica tiene un consumo aproximado de 1.75 A y también porque admite ya sea voltaje AC como DC.

### 3.5 Análisis para determinar la calidad del proyecto (Objetivo 6)

Para determinar la calidad del producto se aplica el estándar ISO 9126 debido a que el proyecto casi en su totalidad es software y el estándar mencionado ha sido desarrollado en un intento de identificar los atributos clave de calidad para el software (evalúa los productos de software), esta norma nos indica las características de la calidad y los lineamientos para su uso.

Para realizar la validación del sistema se evalúa tres características en particular del estándar mencionado los cuales son la funcionalidad que abarca lo relacionado a la exactitud del sistema, la usabilidad que se refiere a que tan amigable resulta con el usuario el sistema, y la eficiencia que evalúa y obtiene resultados en base a los tiempos de respuesta.

#### 3.5.1 Funcionalidad

El proyecto esta implementado en la casa de uno de los autores pero se lo puede implementar en cualquier lugar que requiera de un control de seguridad. Se utiliza como técnica de recolección de datos la observación experimental ya que se realizaron pruebas en base a la experimentación final del sistema para esto se emplea una hoja de registro cuando el sistema detecta o no detecta al usuario.

El siguiente paso es recolectar la información necesaria para lo cual se considera como aceptable el reconocimiento hasta dos intentos y el tercero se considera como no aceptable. En la siguiente tabla 3-3 se muestra los datos obtenidos para 25 usuarios.

**Tabla 3-3:** Datos obtenidos del funcionamiento del sistema

Usuario	Reconoce		No Reconoce
	Intento 1	Intento 2	
1			X
2		✓	
3		✓	
4	✓		
5	✓		
6		✓	

7	✓		
8			X
9	✓		
10	✓		
11		✓	
12	✓		
13			X
14	✓		
15	✓		
16	✓		
17		✓	
18	✓		
19	✓		
20		✓	
21			X
22		✓	
23	✓		
24	✓		
25	✓		
	14	7	4
<b>Total</b>	0,56	0,28	0,16
<b>Total %</b>	56%	28%	16%
	84%		16%

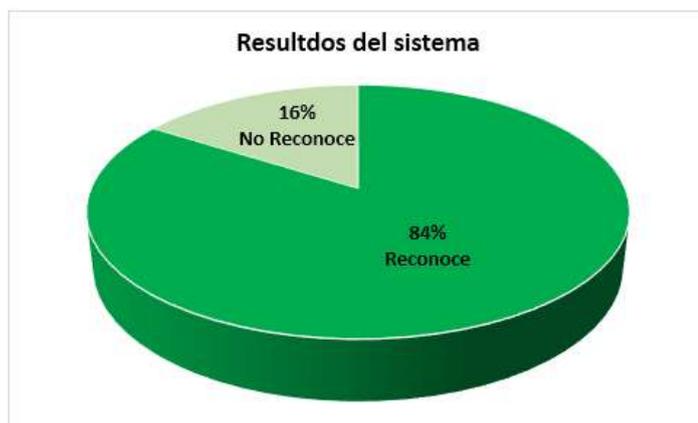
Realizado por: JARA, Caroline; CABA, Carlos 2018

Mediante el análisis experimental realizado al sistema se llegó a la conclusión que existe un 84% de aciertos en el reconocimiento y que el 16% corresponde a pruebas en las cuales no reconoció al tercer intento y se lo consideró como fallo. Se puede observar en el gráfico 3-3 el resultado obtenido del análisis del sistema.

**Tabla 4-3:** Resultados de pruebas de funcionamiento

<b>Resultados de las pruebas realizadas</b>		
	<b>Reconoce</b>	<b>No Reconoce</b>
<b>Total %</b>	84%	16%

Realizado por: JARA, Caroline; CABA, Carlos 2018



**Gráfico 3-3:** Porcentaje de funcionamiento del sistema  
 Realizado por: JARA, Caroline; CABA, Carlos 2018

### 3.5.2 Usabilidad

Para determinar la usabilidad del proyecto se realizó observaciones en tres situaciones distintas las cuales consistieron en proceso de registro de usuarios, manejo del sistema y proceso de petición de acceso al sistema. El objetivo de las observaciones es determinar el porcentaje de ayuda que necesitan las personas para desenvolverse en los tres ámbitos que se está evaluando.

A continuación en la tabla 5-3 y el gráfico 4-3 se muestra los porcentajes de ayuda que necesitaron los usuarios para el proceso de registro al sistema.

**Tabla 5-3:** Porcentaje de ayuda en proceso de registro

Indicador	Proceso de registro	Porcentaje %
Ninguna	15	60%
Pocas	8	32%
Muchas	2	8%
<b>Total</b>	<b>25</b>	<b>100%</b>

Realizado por: JARA, Caroline; CABA, Carlos 2018



**Gráfico 4-3:** Porcentaje de ayuda en proceso de registro  
**Realizado por:** JARA, Caroline; CABA, Carlos 2018

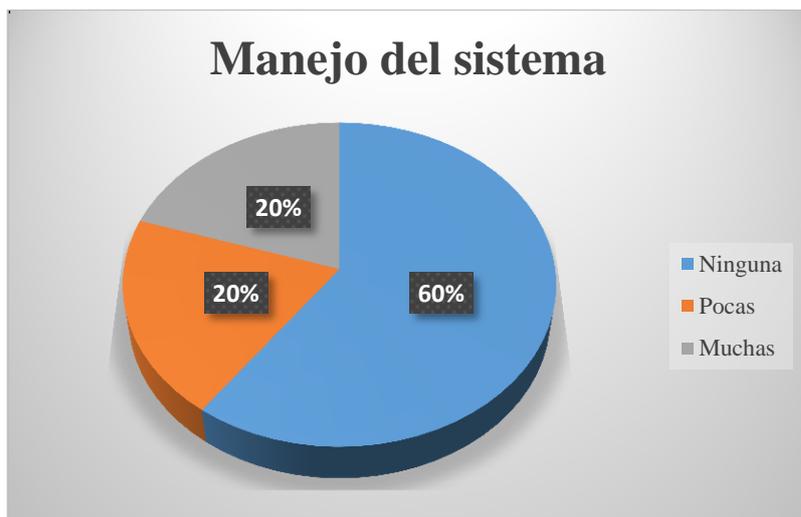
Como podemos observar en la tabla del total de pruebas, 15 personas no necesitaron de indicaciones previas para realizar el registro de usuarios lo que corresponde al 60 %, 8 personas pidieron una explicación en ciertos puntos un poco confusos lo que corresponden al 32%, por ultimo para 2 personas resulto difícil ejecutar el proceso de registro para lo que se les tuvo que realizar una demostración previa, esto corresponde al 8%.

A continuación en la tabla 6-3 y el gráfico 5-3 se muestra los porcentajes de ayuda que necesitaron los usuarios para el manejo del sistema.

**Tabla 6-3:** Porcentaje de ayuda en manejo del sistema

Indicador	Manejo del sistema	Porcentaje %
<b>Ninguna</b>	15	60%
<b>Pocas</b>	5	20%
<b>Muchas</b>	5	20%
<b>Total</b>	25	100%

**Realizado por:** JARA, Caroline; CABA, Carlos 2018



**Gráfico 5-3:** Porcentaje de ayuda en manejo del sistema  
**Realizado por:** JARA, Caroline; CABA, Carlos 2018

Como se puede observar en la tabla 6-3 el número de personas que pudieron manejar el sistema sin indicaciones previas corresponden al 60 %, es decir 15 personas. Personas que solamente necesitaron una pequeño indicación para manejar el sistema corresponden al 20 % y el restante 20 % corresponde a personas a las cuales se les tuvo que dar una demostración.

A continuación en la tabla 7-3 y el gráfico 6-3 se muestra los porcentajes de ayuda que necesitaron los usuarios para solicitar acceso al sistema.

**Tabla 7-3:** Porcentaje de ayuda en petición de acceso

Indicador	Petición de ingreso	Porcentaje %
<b>Ninguna</b>	8	32%
<b>Pocas</b>	12	48%
<b>Muchas</b>	5	20%
<b>Total</b>	25	100%

**Realizado por:** JARA, Caroline; CABA, Carlos 2018



**Gráfico 6-3:** Porcentaje de ayuda en petición de acceso  
 Realizado por: JARA, Caroline; CABA, Carlos 2018

Como se puede ver en la tabla 7-3 en el proceso de petición de acceso o ingreso el 32% de las personas no necesitaron de indicaciones, el 48% solamente tenía ciertas dudas de cómo ejecutar dicha acción y en cambio el 20% no supieron como pedir acceso al sistema y se los tuvo que dar una capacitación.

### 3.5.3 Eficiencia

Para determinar la eficiencia del proyecto se realizó observaciones de los tiempos de ejecución en tres situaciones distintas las cuales consistieron en registro al sistema, reconocimiento facial y creación del modelo facial 3D. El objetivo de las observaciones es determinar el tiempo de respuesta del sistema para las actividades mencionadas.

Se consideró como tiempo de espera tres rangos los cuales se describen en la tabla 8-3.

**Tabla 8-3:** Rangos tiempo de espera

Tiempo en (s)	
<b>Bajo</b>	1 - 20
<b>Medio</b>	21 - 40
<b>Alto</b>	41 - 60

Realizado por: JARA, Caroline; CABA, Carlos 2018

A continuación en la tabla 9-3 y el gráfico 7-3 se muestra los porcentajes de tiempo de los usuarios para el proceso de registro al sistema.

**Tabla 9-3:** Porcentaje de tiempo de respuesta

<b>Indicador</b>	<b>Tiempo de respuesta</b> <b>Registro al sistema</b>	<b>Porcentaje</b> <b>%</b>
<b>Bajo</b>	6	24%
<b>Medio</b>	13	52%
<b>Alto</b>	6	24%
<b>Total</b>	<b>25</b>	<b>100%</b>

Realizado por: JARA, Caroline; CABA, Carlos 2018



**Gráfico 7-3:** Porcentaje de respuesta para el registro al sistema

Realizado por: JARA, Caroline; CABA, Carlos 2018

Como podemos observar en la tabla del total 9-3, 6 personas consideran que el tiempo en que se demoran en registrarse al sistema es bajo lo que corresponde al 24 %, 13 personas consideran que el tiempo es medio correspondiente al 52%, por último 6 personas consideran un tiempo alto esto corresponde al 24%. A continuación en la tabla 10-3 y el gráfico 8-3 se muestra los porcentajes de tiempo de respuesta al reconocimiento facial del sistema.

**Tabla 10-3:** Porcentaje de tiempo de respuesta

<b>Indicador</b>	<b>Tiempo de respuesta</b> <b>Reconocimiento facial</b>	<b>Porcentaje</b> <b>%</b>
<b>Bajo</b>	11	44%
<b>Medio</b>	9	36%
<b>Alto</b>	5	20%
<b>Total</b>	<b>25</b>	<b>100%</b>

Realizado por: JARA, Caroline; CABA, Carlos 2018



**Gráfico 8-3:** Porcentaje de respuesta al reconocimiento facial del sistema  
**Realizado por:** JARA, Caroline; CABA, Carlos 2018

Como podemos observar en la tabla del total 10-3, 11 personas consideran que el tiempo en que se demora en ejecutar la parte de reconocimiento facial del sistema es bajo lo que corresponde al 44%, 9 personas consideran que el tiempo es medio correspondiente al 36%, por último 5 personas consideran un tiempo alto esto corresponde al 20%.

A continuación en la tabla 11-3 y el gráfico 9-3 se muestra los porcentajes de tiempo de respuesta de la creación del modelo facial 3D del sistema.

**Tabla 11-3:** Porcentaje de tiempo de respuesta

Indicador	Tiempo de respuesta Creación modelo 3D	Porcentaje %
<b>Bajo</b>	8	32%
<b>Medio</b>	13	52%
<b>Alto</b>	4	16%
<b>Total</b>	<b>25</b>	<b>100%</b>

**Realizado por:** JARA, Caroline; CABA, Carlos 2018



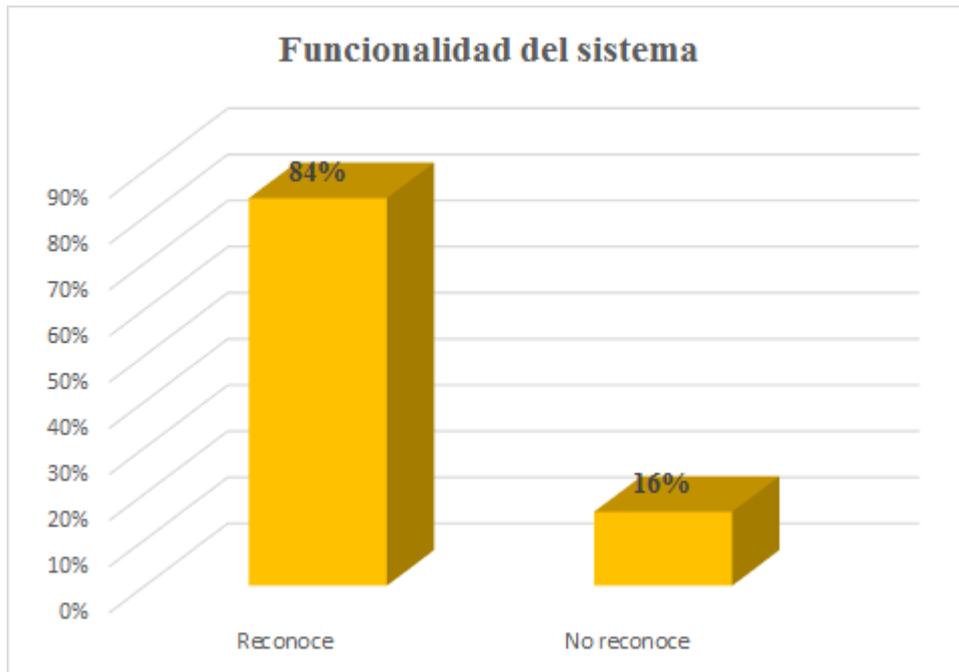
**Gráfico 9-3:** Porcentaje de respuesta del modelo facial 3D del sistema  
Realizado por: JARA, Caroline; CABA, Carlos 2018

Como podemos observar en la tabla del total 11-3, 8 personas consideran que el tiempo en que se demora en ejecutar la parte del modelo facial 3D del sistema es bajo lo que corresponde al 32%, 13 personas consideran que el tiempo es medio correspondiente al 52%, por último 4 personas consideran un tiempo alto esto corresponde al 16%.

### 3.6 Discusión de resultados.

Para determinar el método de detección de rostro que se implemente se realizó una comparación entre los métodos de Viola Jones y el modelo unificado, esta comparación se basó en los parámetros de tiempo de detección, posiciones detectadas, variación en la iluminación, y variación en la distancia a la cámara. Se determinó que el mejor método de detección para la elaboración del proyecto es el modelo unificado debido a que brinda un 42 % del total de intentos, es decir necesita menor número de intentos para realizar la detección ante variaciones en los parámetros de captura, y brinda el 83% de las detecciones en varias posiciones y con un tiempo similar al del método de Viola Jones.

En funcionalidad como se puede ver en la gráfica 10-3 el sistema muestra un 84% de reconocimientos exitosos bajo la condición que tenga un máximo de dos intentos para realizarlo, por otra parte para el 16% de personas el sistema fallo en el intento de reconocimiento y se pudo observar que los intentos fallidos fueron con personas de baja estatura lo que provocó que las tomas de imágenes no se realizaran de forma centrada y en ocasiones no lograron capturarse los rostros en la fotografía.



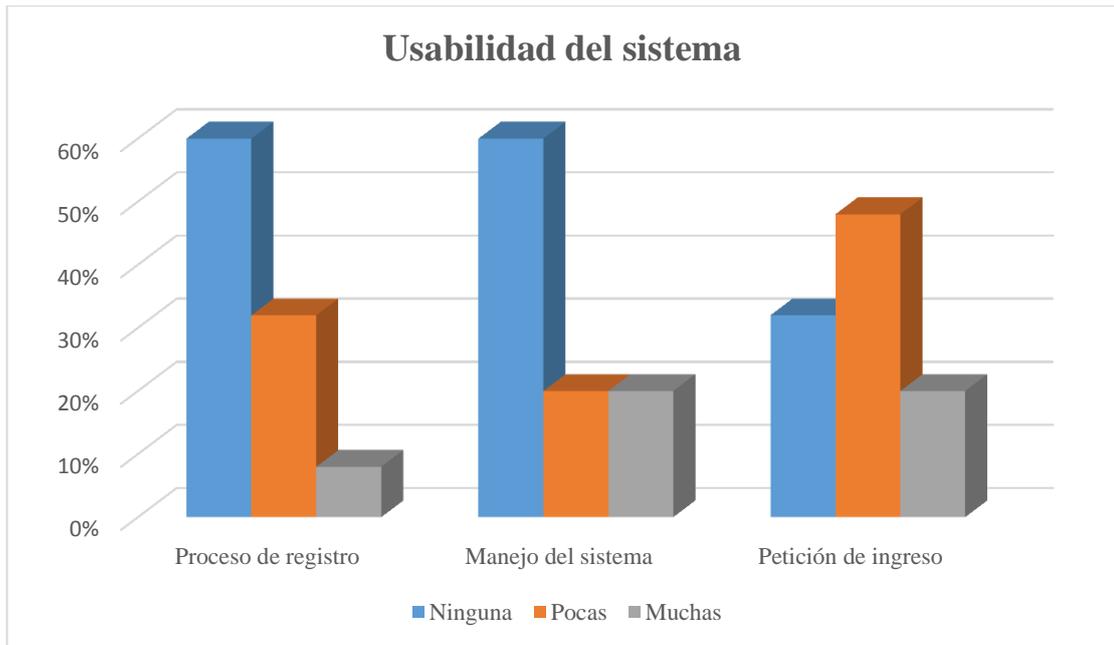
**Gráfico 10-3:** Funcionalidad del sistema  
**Realizado por:** JARA, Caroline; CABA, Carlos 2018

En usabilidad como se puede ver en la gráfica 11-3 el sistema obtuvo un 84% de personas que lograron manejar correctamente el sistema debido a que la interface así lo permitió, por otra parte 16% de personas no fueron capaces de interactuar con el sistema sin la necesidad de recibir capacitación alegando que no han manejado con anterioridad sistemas similares.

**Tabla 12-3:** Resumen de usabilidad del sistema.

Indicador	Proceso de registro	Manejo del sistema	Petición de ingreso
<b>Ninguna</b>	60%	60%	32%
<b>Pocas</b>	32%	20%	48%
<b>Muchas</b>	8%	20%	20%
<b>Total</b>	100%	100%	100%

**Realizado por:** JARA, Caroline; CABA, Carlos 2018



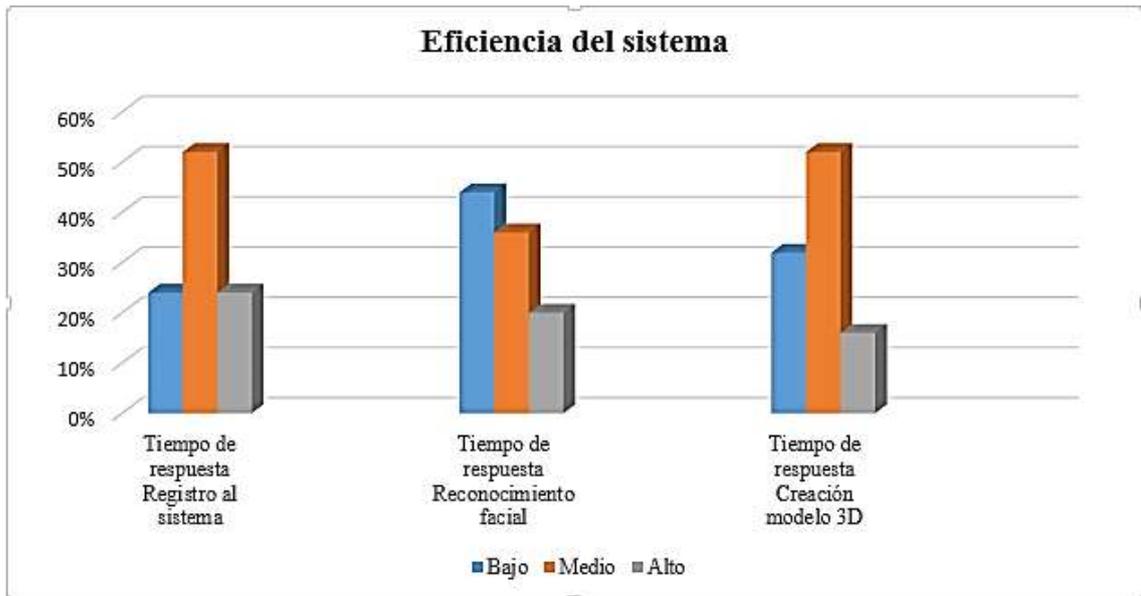
**Gráfico 11-3:** Usabilidad del sistema  
**Realizado por:** JARA, Caroline; CABA, Carlos 2018

En cuanto a eficiencia como se puede ver en la gráfica 12-3 se obtuvo que 80% de personas consideran que el tiempo de respuesta es adecuado para este tipo de sistemas, este grupo de personas comprendieron el hecho de usar reconocimiento por parte doble (imagen de frente e imagen de perfil) provoca que el tiempo tiende a extenderse, por otra parte el 20% de personas indicaron que el sistema responde de forma demorada para el objetivo que fue diseñado.

**Tabla 13-3:** Resumen de eficiencia del sistema.

Indicador	Tiempo de respuesta Registro al sistema	Tiempo de respuesta Reconocimiento facial	Tiempo de respuesta Creación modelo 3D
<b>Bajo</b>	24%	44%	32%
<b>Medio</b>	52%	36%	52%
<b>Alto</b>	24%	20%	16%
<b>Total</b>	<b>100%</b>	<b>100%</b>	<b>100%</b>

**Realizado por:** JARA, Caroline; CABA, Carlos 2018



**Gráfico 12-3:** Eficiencia del sistema  
 Realizado por: JARA, Caroline; CABA, Carlos 2018

## CONCLUSIONES

Se implementó un sistema de reconocimiento aplicado a la seguridad utilizando el algoritmo de reconocimiento EBGM y 3DMM para la creación del modelo facial; utilizando un algoritmo de inteligencia artificial se permite el acceso sin la probabilidad de algún error de identificación, debido a que muchas veces un rasgo en particular de una persona suelen ser idéntico al de otras personas, pero no su rostro en total.

Se elaboró una base de datos en el software MySQL la cual contiene la información de los usuarios, personas admitidas y personas rechazadas del sistema, y se logró realizar la conexión con el entorno de Matlab para la gestión de la base mencionada.

Se logró realizar la comunicación entre Matlab, arduino y los elementos de entrada y salida del sistema para tener un trabajo conjunto el cual controló el acceso de usuarios a un determinado lugar mediante la acción de una cerradura controlada automáticamente, y se implementó la creación del modelo facial 3D para aquellas personas que sin estar registradas en la base de datos hayan solicitado acceso.

Para la elaboración del sistema se realizó un previo análisis estadístico de los métodos de reconocimiento facial y se determinó que el modelo unificado es el mejor para los objetivos planteados.

Se logró realizar las pruebas correspondientes en base a los parámetros estipulados en la norma ISO 9126 y se determinó que el sistema cuenta con un 84% de funcionalidad, 84% de usabilidad y con un 80% de eficiencia.

## **RECOMENDACIONES**

Se recomienda para la instalación de cámaras web utilizar extensiones de USB máximo de tres metros ya que de lo contrario empieza a existir pérdida de información y la posición de las cámaras para este sistema deben ser colocadas de forma equidistante y perpendicular entre sí para cumplir con los requisitos del proyecto.

Se recomienda el uso de la placa diseñada e implementada para la protección del equipo portátil, o el arduino de posibles fallos por sobre voltajes, más aun cuando se trabaja en el circuito de mando con voltajes elevados en relación a los permitidos por dichos dispositivos.

Al trabajar con imágenes y con bases de datos, lo recomendable es utilizar la base para almacenar únicamente la dirección en la cual se encuentra la imagen guardada, ya que intentar almacenar todo el contenido de una imagen en una tabla produciría un excesivo uso de memoria y ralentizaría el proceso.

Si llegara a elaborarse algún otro algoritmo de reconocimiento es recomendable tener muy en cuenta el tiempo de respuesta del mismo ya que al tratarse de seguridad automatizada los aspectos importantes son la confiabilidad y la eficiencia del sistema.

## BIBLIOGRAFÍA

*Banco Guayaquil lanza sistema de reconocimiento facial para hacer transacciones* [En línea]. Ecuador: Tapia, 2015. [Consulta 5 noviembre 2017]. Disponible en:  
<http://www.elcomercio.com/actualidad/banco-guayaquil-lanza-sistema-reconocimiento.html>

**BLÁZQUEZ PÉREZ, Luis.** *Reconocimiento Facial Basado en Puntos Característicos de la Cara en entornos no controlados* (Tesis). (Pregrado) Escuela Politécnica Superior Universidad Autónoma de Madrid, Dpto. de Ingeniería Informática. Madrid. 2013. pp. 6. [Consulta: 15 diciembre 2017]. Disponible en: [http://audias.ii.uam.es/seminars/PFC\\_Luis\\_Blazquez.pdf](http://audias.ii.uam.es/seminars/PFC_Luis_Blazquez.pdf)

**GARCÍA MENDOZA, Consuelo Varinia.** *Modelado de software con lógica difusa* (Tesis). (Maestría) Instituto Politécnico Nacional. México Distrito Federal. 2006. pp. 11-12. [Consulta: 18 agosto 2017]. Disponible en:  
[http://repositoriodigital.ipn.mx/bitstream/123456789/10714/1/PTA\\_M\\_20061214\\_001.PDF](http://repositoriodigital.ipn.mx/bitstream/123456789/10714/1/PTA_M_20061214_001.PDF)

**GONZÁLES MARTÍN, Sara.** Desarrollo de una herramienta para analizar métodos de reconocimiento de emociones (Tesis). (Pregrado) Escuela Técnica Superior de Ingeniería y Sistemas de Telecomunicación, Dpto. Teoría de la Señal y Comunicaciones. Madrid. 2015. pp 12-13. [Consulta: 6 junio 2017]. Disponible en:  
[http://oa.upm.es/39570/8/TFG\\_SARA\\_GONZALEZ\\_MARTIN.pdf](http://oa.upm.es/39570/8/TFG_SARA_GONZALEZ_MARTIN.pdf)

**GUIMARAES NUNES, Cristiano Fraga.** *Descriptor local basado en filtros log-gabor para correspondencia de puntos clave entre imágenes multispectrales* (Tesis). (Pregrado) Centro Federal de Educación Tecnológica de Minas Gerais, Área de Modelado Matemático y Equipo. 2017. pp 20-21. [Consulta: 12 enero 2018]. Disponible en:  
[http://piim-lab.decom.cefetmg.br/galerias/publicacoes\\_arquivos/dissertacoes/cristiano-2017.pdf](http://piim-lab.decom.cefetmg.br/galerias/publicacoes_arquivos/dissertacoes/cristiano-2017.pdf)

**INTRONA, L., & NISSENBAUM, H.** *Facial Recognition Technology a Survey of Policy and Implementation Issues* [En línea]. Lancaster University Management School, the Department of Organisation, Work and Technology. 2010. pp. 15-16. [Consulta: 7 julio 2017]. Disponible en:  
<http://eprints.lancs.ac.uk/49012/1/Document.pdf>

**MARTÍNEZ CUENCA, Diego Fernando.** *Procesamiento de imágenes a partir de muestras de imágenes provistas de una cámara en tiempo real utilizando visión artificial* (Tesis). (Pregrado) Universidad Politécnica Salesiana. Quito-Ecuador. 2016. pp. 12. [Consulta: 20 diciembre 2017]. Disponible en:

<https://dspace.ups.edu.ec/bitstream/123456789/13043/1/UPS%20-%20ST002238.pdf>

*Objetivos Nacionales para el Buen Vivir* [En línea]. Ecuador: Desarrollo, 2018. [Consulta: 26 noviembre 2017]. Disponible en: <http://www.buenvivir.gob.ec/objetivo-10.-impulsar-la-transformacion-de-la-matriz-productiva#tabs2>

**OROPEZA ORTEGA, Juan Carlos.** *Reconocimiento de rostros* (Tesis). (Maestría) Instituto Politécnico Nacional, Escuela Superior De Ingeniería Mecánica Y Eléctrica, Sección de Estudios de Posgrado e Investigación Unidad Culhuacán. México, D.F. 2008. pp. 68-70. [Consulta: 15 noviembre 2017]. Disponible en: <http://tesis.ipn.mx/handle/123456789/1210>

**PEREYRA, Pamela Alejandra.** *Reconocimiento Facial Mediante Imágenes Estereoscópicas Para Control de Ingreso* (Tesis). (Pregrado) Universidad De Buenos Aires, Facultad De Ingeniería. Buenos Aires. pp. 44-46. [Consulta: 4 diciembre 2017]. Disponible en: <http://materias.fi.uba.ar/7500/PamelaPereira.pdf>

**PEREZ ORR, Julio Daniel.** *Análisis de Imágenes Faciales basado en Propiedades Espectrales de Grafos* (Tesis). (Pregrado) Universidad Carlos III de Madrid, Escuela Politécnica Superior, Ingeniería Superior de Telecomunicación. Leganés. 2009. pp 32-33. [Consulta: 16 julio 2017]. Disponible en: [https://e-archivo.uc3m.es/bitstream/handle/10016/5910/PFC\\_Daniel\\_Perez\\_Orr.pdf?sequence=1](https://e-archivo.uc3m.es/bitstream/handle/10016/5910/PFC_Daniel_Perez_Orr.pdf?sequence=1)

**SALAZAR ESPINOZA, Christian F.** *Diseño e Implementación de un sistema de seguridad para un automóvil con autenticación por reconocimiento facial utilizando técnicas de visión artificial* (Tesis). (Pregrado) Escuela Superior Politécnica de Chimborazo, Facultad de Informática y Electrónica, Escuela de Ingeniería Electrónica en Control y Redes Industriales. Riobamba-Ecuador. 2016. pp. 52-62. [Consulta: 9 enero 2018]. Disponible en: <http://dspace.esPOCH.edu.ec/handle/123456789/5008>

**SANTANA JIMÉNEZ, Lisette.** *Una aplicación de la lógica difusa a la evaluación del balance de riesgos de la inflación y del crecimiento macroeconómico* [En línea]. Santo Domingo- República Dominicana. 2013, pp. 497-514. [Consulta: 16 septiembre 2017]. Disponible en: <http://www.redalyc.org/articulo.oa?id=87028918004>

**VALENCIA, G., & BRITO, V.** *Implementación de un control de acceso con reconocimiento facial mediante OpenCV para la sala de Profesores de la Fie* (Tesis). (Pregrado) Escuela

Superior Politécnica de Chimborazo, Facultad de Informática y Electrónica, Escuela de Ingeniería Electrónica en Control y Redes Industriales. Riobamba-Ecuador. 2015. pp. 42-55. [Consulta: 15 diciembre 2017]. Disponible en <http://dspace.esPOCH.edu.ec/handle/123456789/5020>

**VILLALÓN DE LA VEGA, Darío Eduardo.** *Diseño e implementación de una plataforma de software para Reconocimiento Facial en video.* (Tesis). (Pregrado) Universidad de Chile, Facultad de Ciencias Física y Matemáticas, Departamento de Ingeniería Eléctrica. Santiago de Chile. 2012. pp. 18-22. [Consulta: 21 agosto 2017]. Disponible en: [http://repositorio.uchile.cl/bitstream/handle/2250/112271/cf-villalon\\_dd.pdf?sequence=1](http://repositorio.uchile.cl/bitstream/handle/2250/112271/cf-villalon_dd.pdf?sequence=1)

**ZAPATERO OLMEDILLO, Diego.** Herramienta de Reconocimiento Facial de Emociones en Android (Tesis). (Pregrado) Universidad Politécnica de Madrid, Escuela Técnica Superior de Ingeniería y Sistemas de Telecomunicación, Dpto. Teoría de la Señal y Comunicaciones. Madrid. 2016. pp 8-10. [Consulta: 27 noviembre 2017]. Disponible en: [http://oa.upm.es/44722/3/TFG\\_DIEGO\\_ZAPATERO\\_OLMEDILLO.pdf](http://oa.upm.es/44722/3/TFG_DIEGO_ZAPATERO_OLMEDILLO.pdf)

**ZHU, X., & RAMANAN, D.** *Face Detection, Pose Estimation, and Landmark Localization in the Wild* [En línea]. University of California, Irvine, Dept. of Computer Science. California. 2012, pp. 1-2. [Consulta: 3 julio 2017]. Disponible en: <https://www.ics.uci.edu/~xzhu/paper/face-cvpr12.pdf>

## ANEXOS

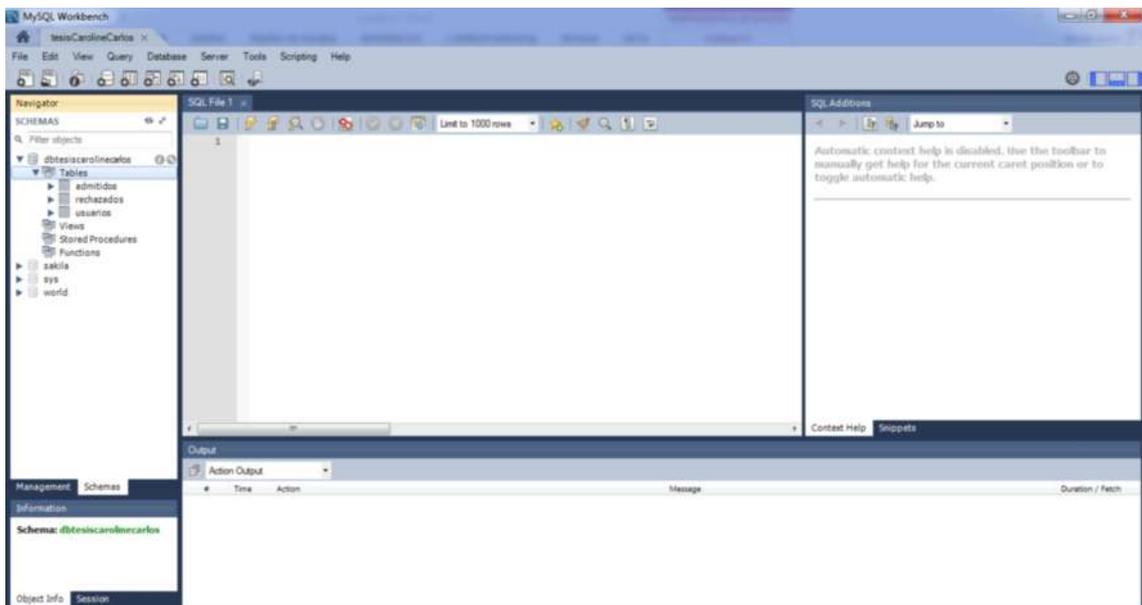
### ANEXO A

## CONEXIÓN MATLAB Y MYSQL

Primero debemos crear nuestra base de datos en MySQL Workbench



**Figura 1-A:** Creación base de datos.  
Realizado por: JARA, Caroline; CABA, Carlos 2018



**Figura 2-A:** Creación base de datos.  
Realizado por: JARA, Caroline; CABA, Carlos 2018

Una vez echo la base de datos, para poder realizar la conexión de Matlab con MySQL es necesario de un conector llamado “MySQL connector ODBC”.



**Figura 3-A:** MySQL connector ODBC.  
Realizado por: JARA, Caroline; CABA, Carlos 2018

Damos doble clic y damos en ejecutar.



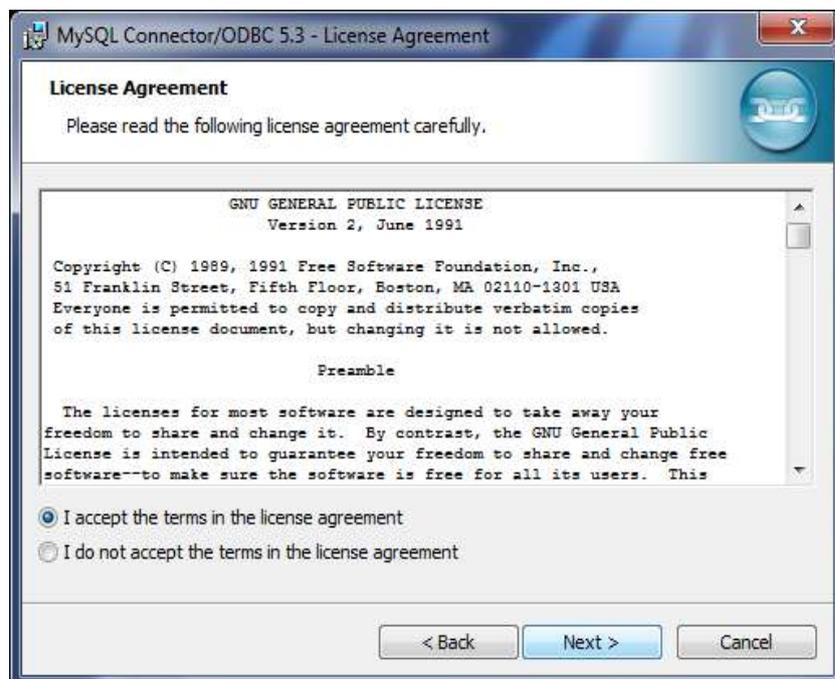
**Figura 4-A:** Instalación del conector ODBC.  
Realizado por: JARA, Caroline; CABA, Carlos 2018

En la ventana que aparece damos en siguiente.



**Figura 5-A:** Instalación del conector ODBC.  
Realizado por: JARA, Caroline; CABA, Carlos 2018

A continuación aceptamos las condiciones y damos en siguiente.



**Figura 6-A:** Instalación del conector ODBC.  
Realizado por: JARA, Caroline; CABA, Carlos 2018

En la siguiente ventana que aparecerá damos en la opción complete y damos a siguiente.



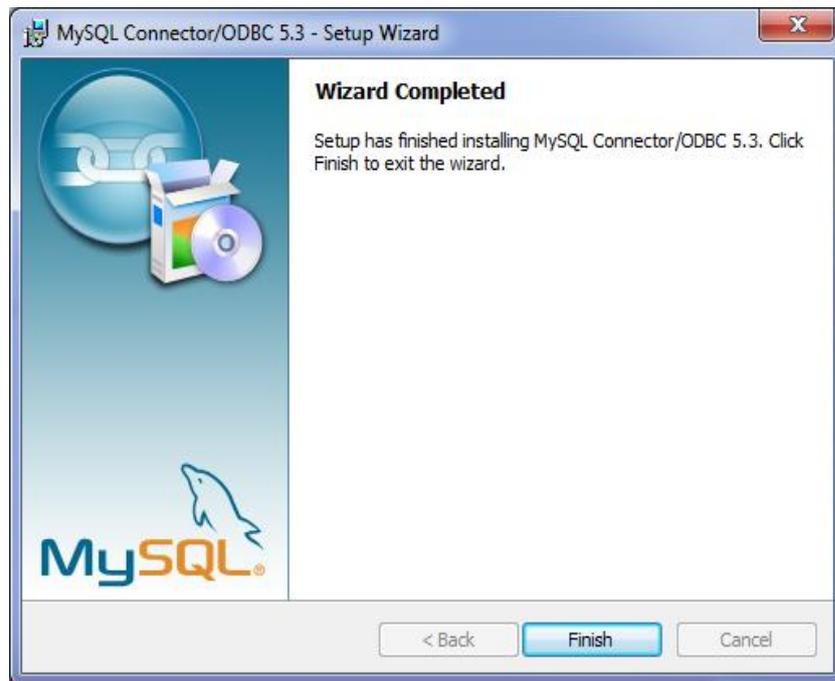
**Figura 7-A:** Instalación del conector ODBC.  
Realizado por: JARA, Caroline; CABA, Carlos 2018

En la siguiente pantalla damos a instalar.



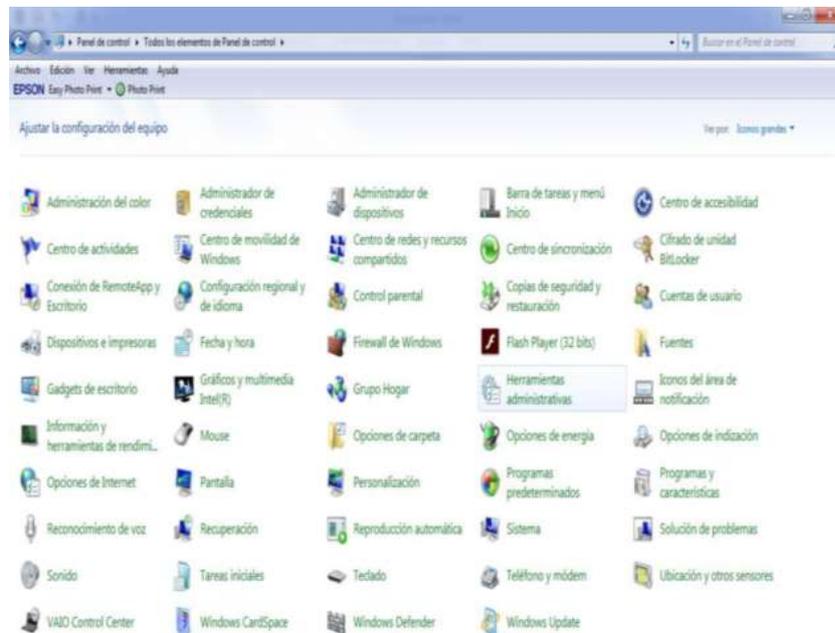
**Figura 8-A:** Instalación del conector ODBC.  
Realizado por: JARA, Caroline; CABA, Carlos 2018

Cuando el proceso haya terminado damos a finalizar.



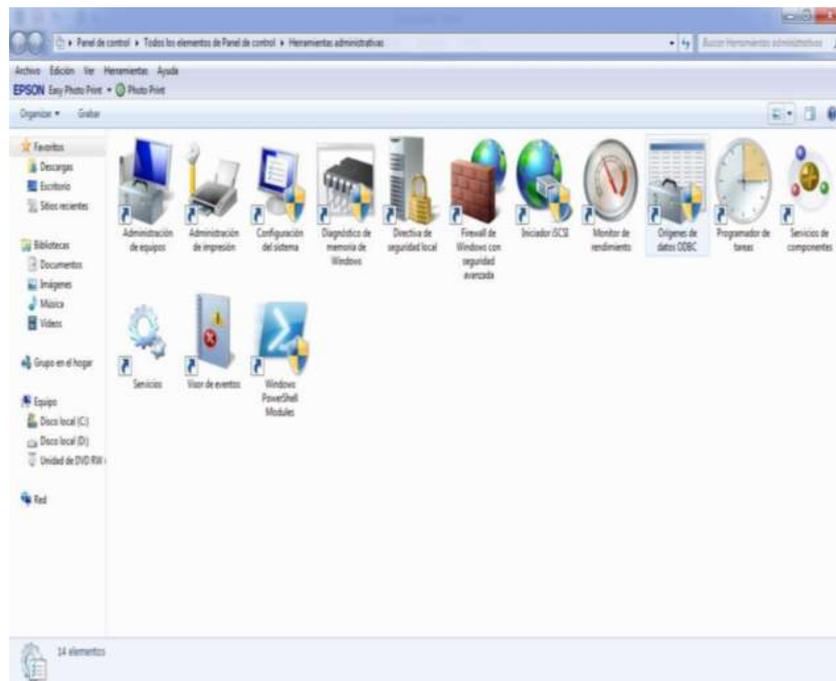
**Figura 9-A:** Instalación del conector ODBC.  
Realizado por: JARA, Caroline; CABA, Carlos 2018

Después nos dirigimos a Panel de control y luego a Herramientas administrativas.



**Figura 10-A:** Instalación de conector ODBC.  
Realizado por: JARA, Caroline; CABA, Carlos 2018

Luego escogemos la opción Orígenes de datos ODBC.

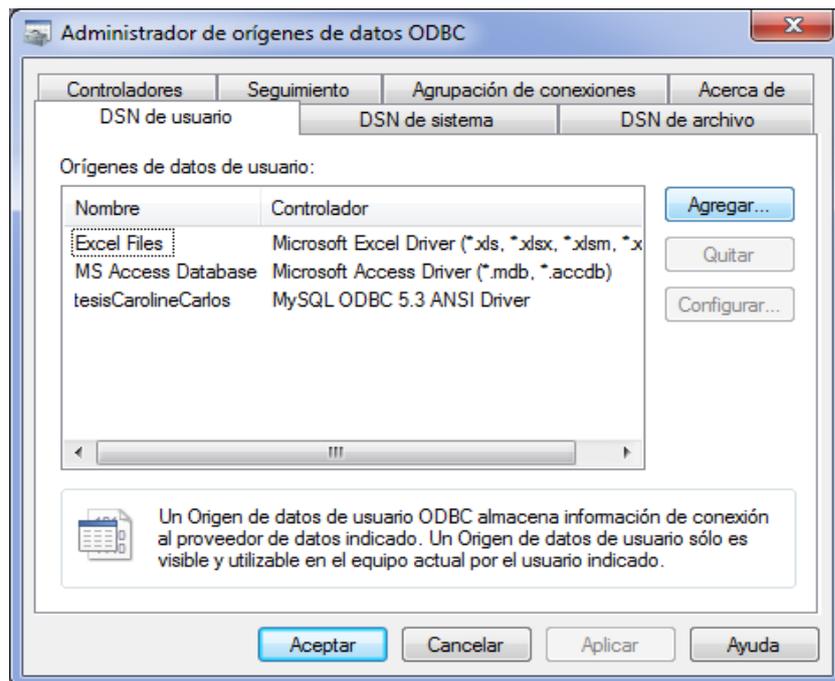


**Figura 11-A:** Instalación del conector ODBC.  
Realizado por: JARA, Caroline; CABA, Carlos 2018



**Figura 12-A:** Instalación del conector ODBC.  
Realizado por: JARA, Caroline; CABA, Carlos 2018

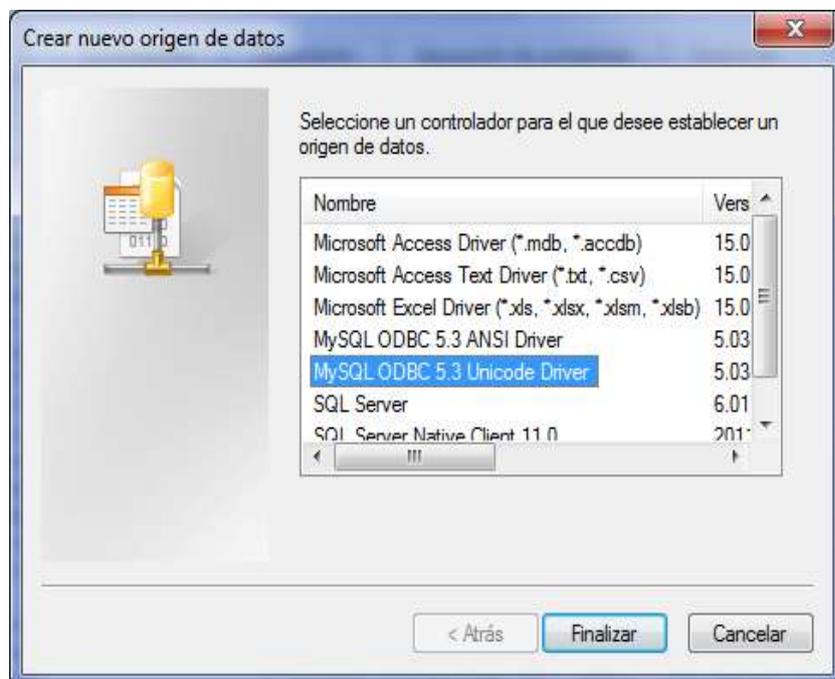
En la ventana que aparece damos en Agregar.



**Figura 13-A:** Instalación del conector ODBC.

Realizado por: JARA, Caroline; CABA, Carlos 2018

En la siguiente ventana desplegada escogemos la opción que se muestra seleccionada y damos en finalizar.



**Figura 14-A:** Instalación del conector ODBC.

Realizado por: JARA, Caroline; CABA, Carlos 2018

En la siguiente ventana que aparece se deben llenar los campos que estén en blanco poniendo un nombre cualquiera, en la parte TCP/IP Server se pone localhost, ponemos el nombre de usuario, si tiene con contraseña la ponemos y abajo seleccionamos la base de datos que creamos en MySQL y damos clic en Test.



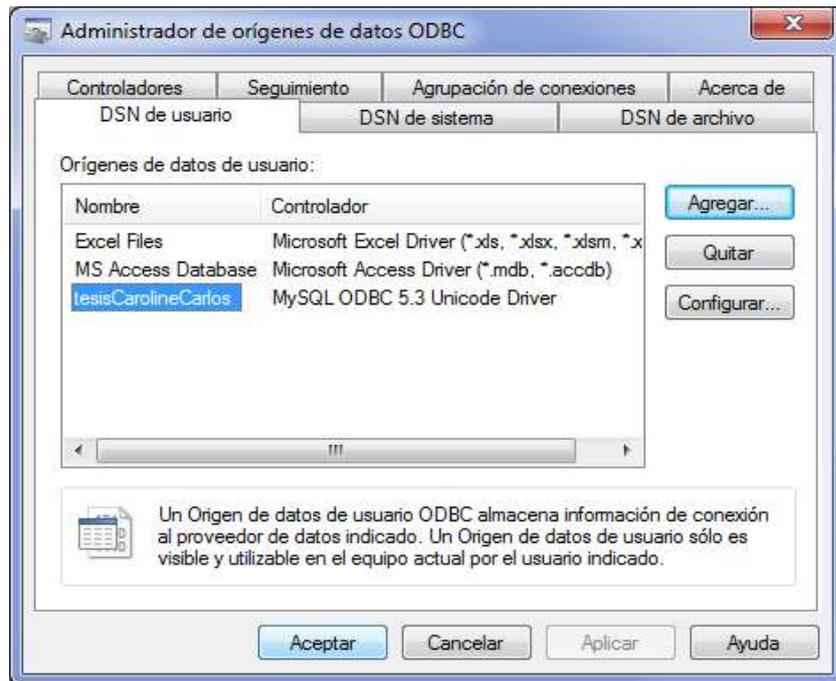
**Figura 15-A:** Conexión MySQL y Matlab.  
Realizado por: JARA, Caroline; CABA, Carlos 2018

Al dar en la opción Test no aparecerá el siguiente mensaje damos en aceptar y en OK.



**Figura 16-A:** Conexión MySQL y Matlab.  
Realizado por: JARA, Caroline; CABA, Carlos 2018

A continuación damos en aceptar.



**Figura 17-A:** Conexión MySQL y Matlab.  
Realizado por: JARA, Caroline; CABA, Carlos 2018

Una vez realizada la base de datos e instalado el conector ODBC nos dirigimos a Matlab y en el comand window escribimos el comando “dexplore” el cual nos desplegara la siguiente ventana



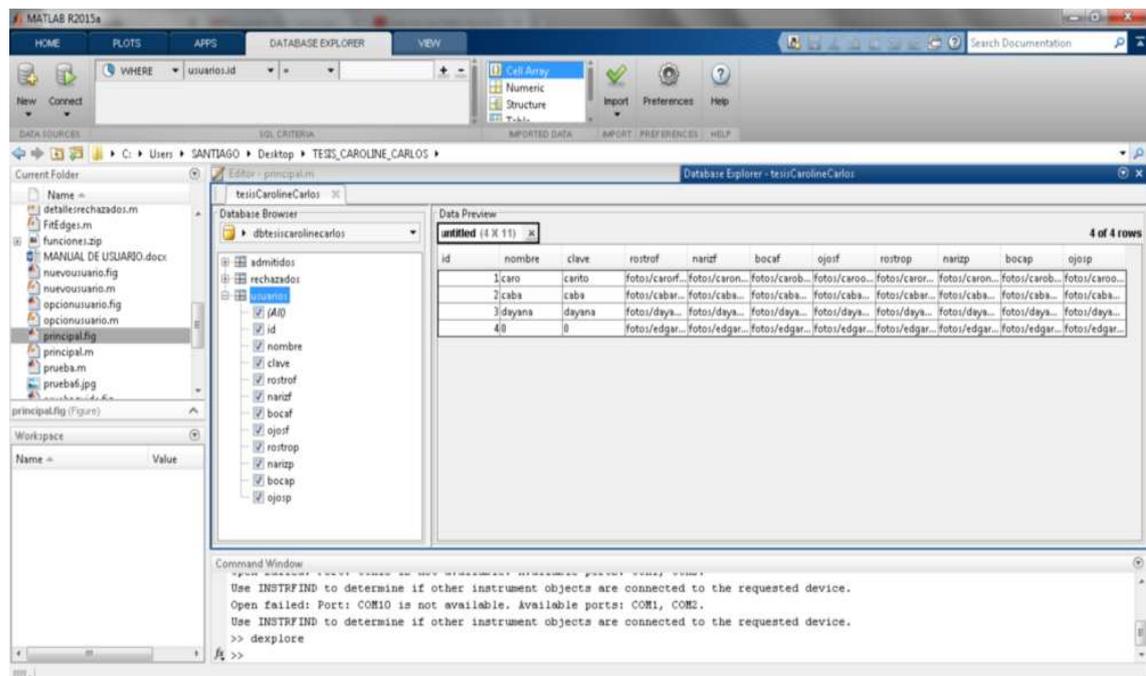
**Figura 18-A:** Conexión MySQL y Matlab.  
Realizado por: JARA, Caroline; CABA, Carlos 2018

Llenamos los campos que nos piden escogemos el nombre de la conexión que hemos hecho anteriormente con el conector ODBC ponemos el nombre de usuario y la contraseña en caso de tenerla y damos en conectar.



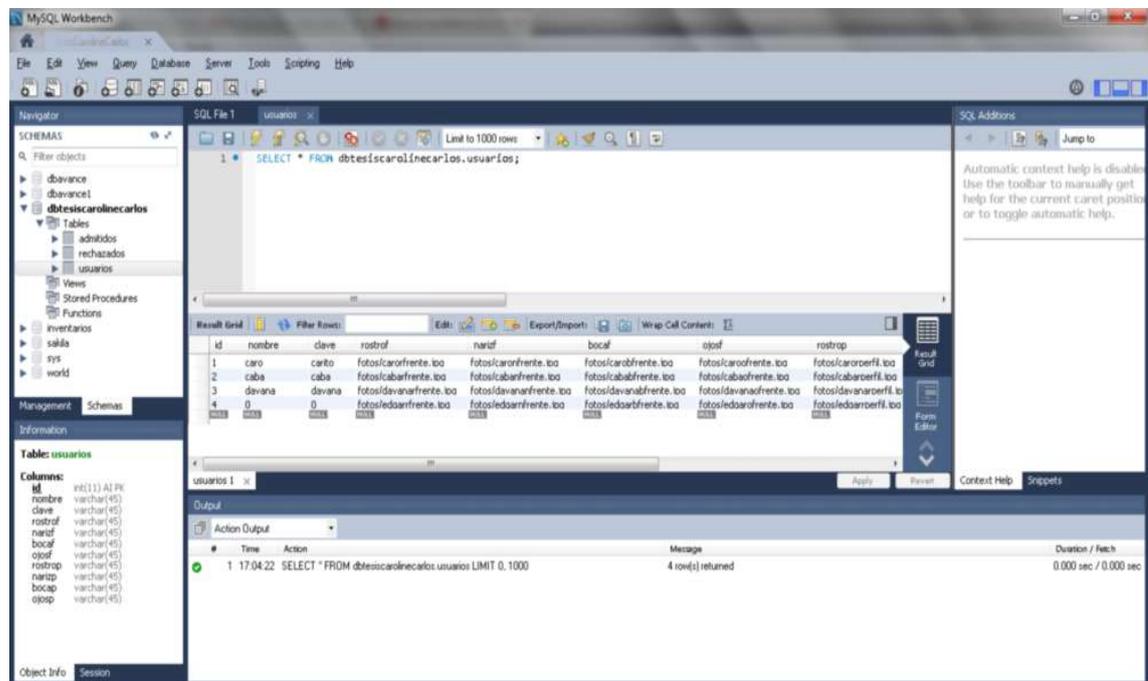
**Figura 19-A:** Conexión MySQL y Matlab.  
Realizado por: JARA, Caroline; CABA, Carlos 2018

Una vez conectado nos aparecera en matlab la base de datos que hemos creado en MySQL ahí vemos las tablas creadas con sus respectivas columnas.



**Figura 20-A:** Conexión MySQL y Matlab.  
Realizado por: JARA, Caroline; CABA, Carlos 2018

En la siguiente imagen podemos ver en MySQL que tenemos las mismas tablas con los mismos usuarios ingresados que en Matlab.

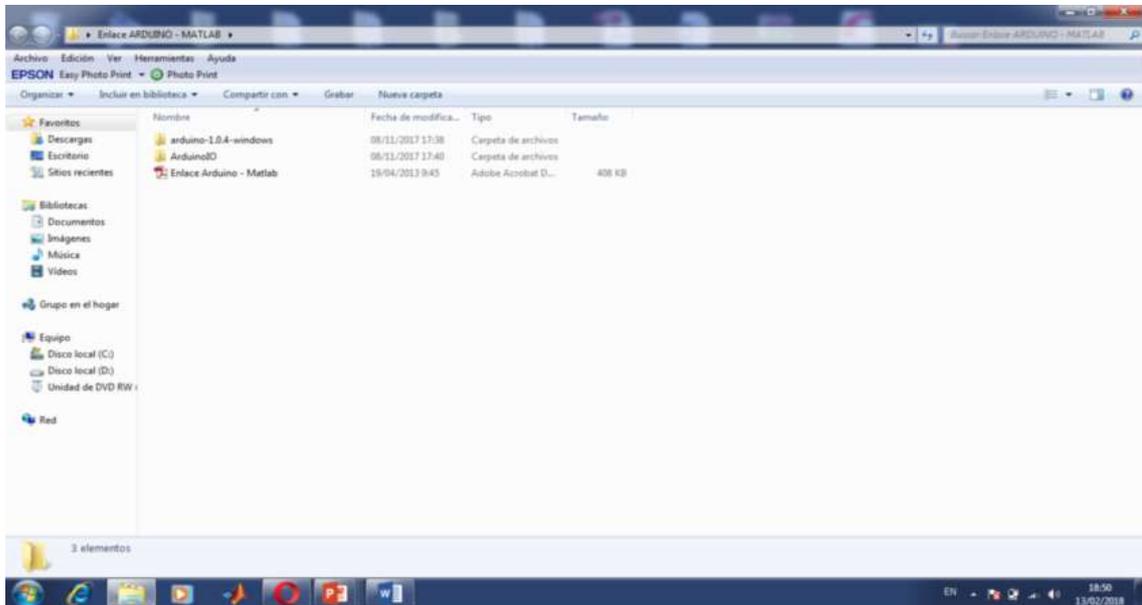


**Figura 21-A:** Verificación de la conexión MySQL y Matlab.  
Realizado por: JARA, Caroline; CABA, Carlos 2018

## ANEXO B

### ENLACE ARDUINO MATLAB

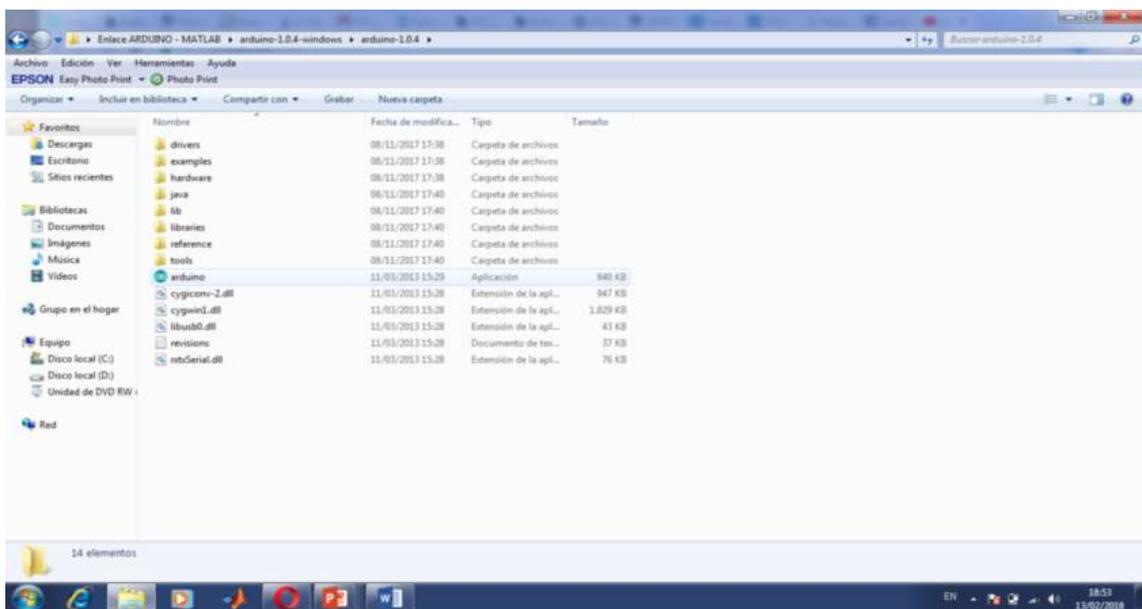
Primero se debe tener descargado Arduino y un paquete desarrollado para MATLAB el cual nos permitirá hacer el enlace.



**Figura 1-B:** Enlace Arduino Matlab.

Realizado por: JARA, Caroline; CABA, Carlos 2018

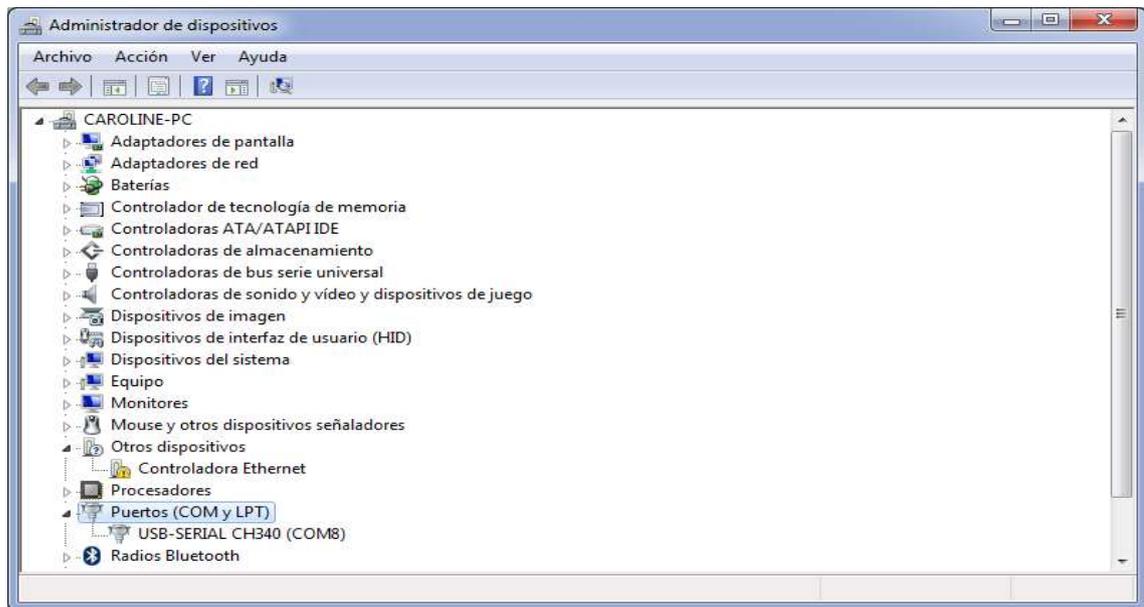
Abrir la carpeta “arduino-1.0.4-windows” luego el “Arduino” y conectar la placa Arduino a través del cable USB.



**Figura 2-B:** Enlace Arduino Matlab.

Realizado por: JARA, Caroline; CABA, Carlos 2018

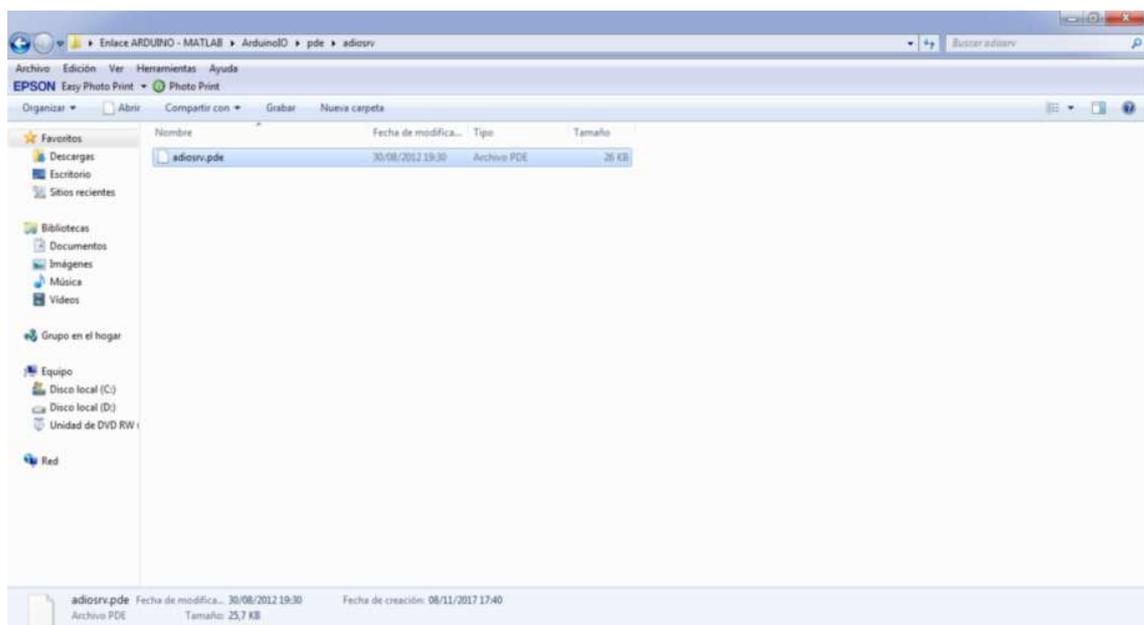
Verificar por cual “COM” quedo conectado. Nos dirigimos a Administrador de dispositivos y Puertos y verificamos el puerto COM.



**Figura 3-B:** Enlace Arduino Matlab.

Realizado por: JARA, Caroline; CABA, Carlos 2018

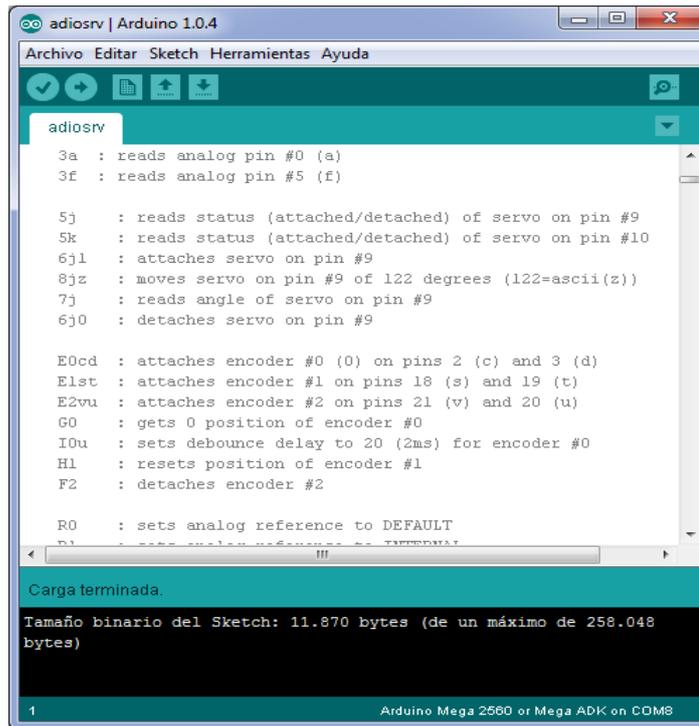
Introducimos un sketch dentro de nuestra placa Arduino la cual nos permitirá hacer la conexión entre el Arduino y Matlab, se le instala este sketch a la placa para que pueda entender las órdenes enviadas desde el MATLAB. C:\Users\CAROLINE\Desktop\Enlace ARDUINO - MATLAB\ArduinoIO\pde\adiosrv



**Figura 4-B:** Enlace Arduino Matlab.

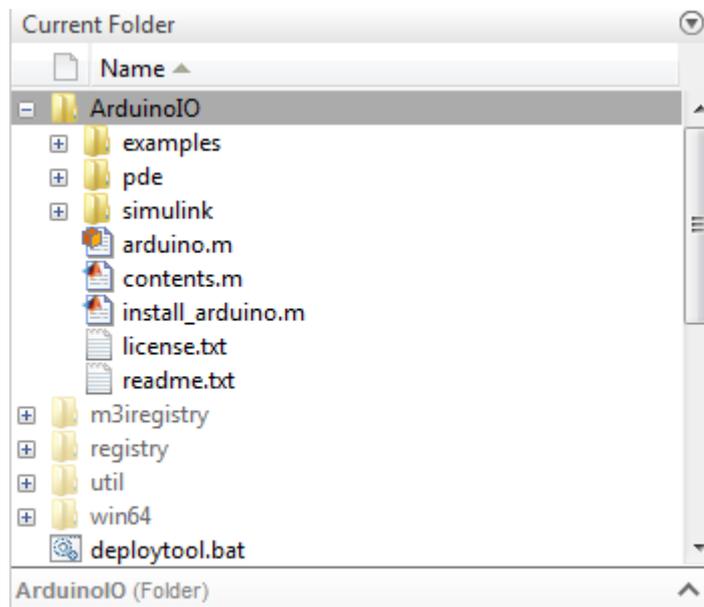
Realizado por: JARA, Caroline; CABA, Carlos 2018

Teniendo el adiosrv abierto le damos a “cargar” y esperamos hasta que diga “Carga terminada.”



**Figura 5-B: Enlace Arduino Matlab.**  
**Realizado por:** JARA, Caroline; CABA, Carlos 2018

Ejecutamos Matlab como administrador y en el directorio “Current Folder” ubicamos la carpeta ArduinoIO.



**Figura 6-B: Enlace Arduino Matlab.**  
**Realizado por:** JARA, Caroline; CABA, Carlos 2018

Digitamos “install\_arduino”.

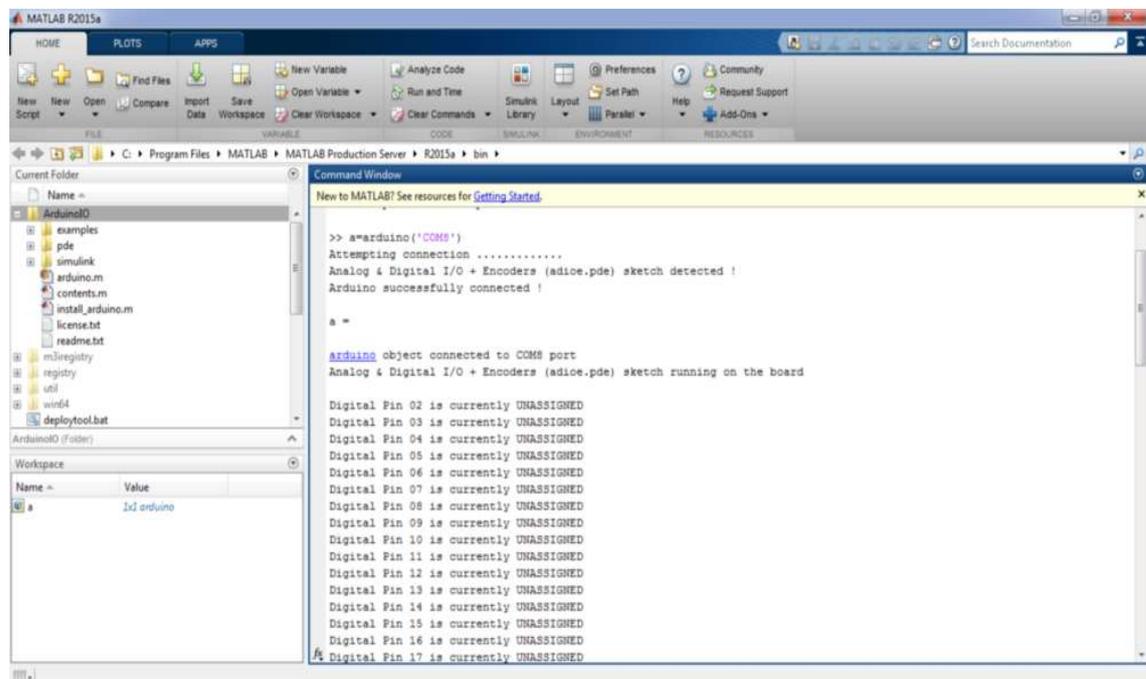
```
Command Window
New to MATLAB? See resources for Getting Started.

>> install_arduino
Warning: There is at least another arduino.m file in the path
to delete any other version before using this one
> In install\_arduino \(line 15\)
  Arduino folders added to the path
  Saved updated MATLAB path
```

**Figura 7-B:** Enlace Arduino Matlab.

Realizado por: JARA, Caroline; CABA, Carlos 2018

Ahora digitamos “a=arduino('COMX)”) Donde X es el número de puerto COM y aparecerá muchas instrucciones. Dichas instrucciones orientaran al usuario en la programación.

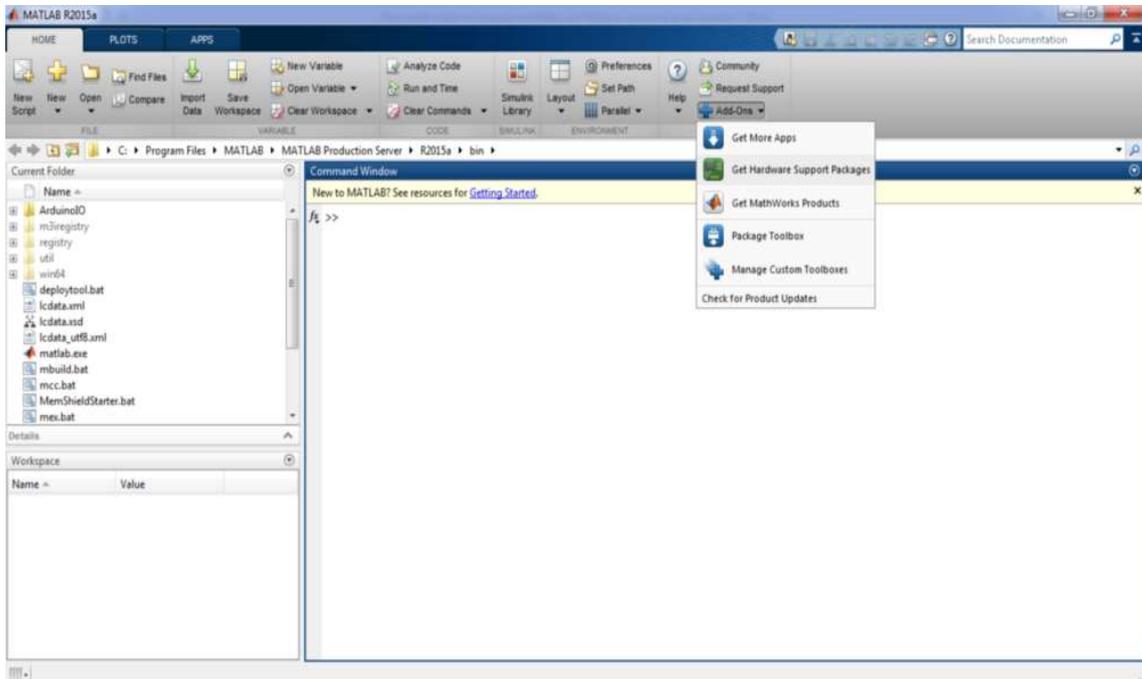


**Figura 8-B:** Enlace Arduino Matlab.

Realizado por: JARA, Caroline; CABA, Carlos 2018

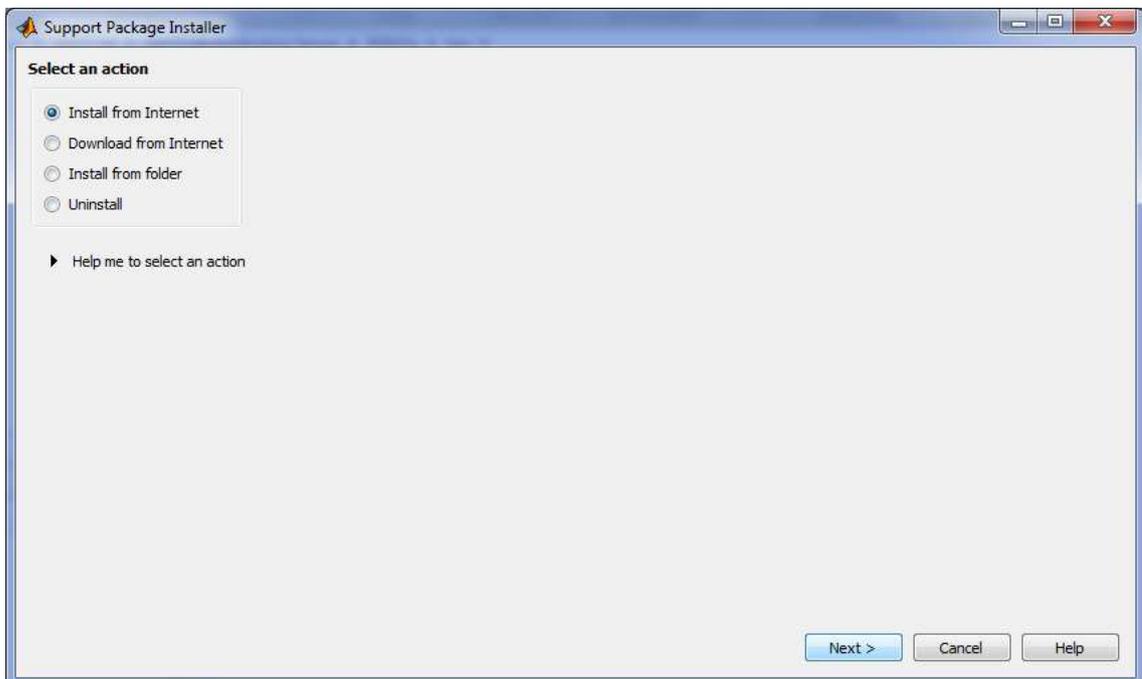
## ANEXO C

### INSTALACIÓN PAQUETES DE MATHWORKS PARA USAR LAS CÁMARAS WEB.



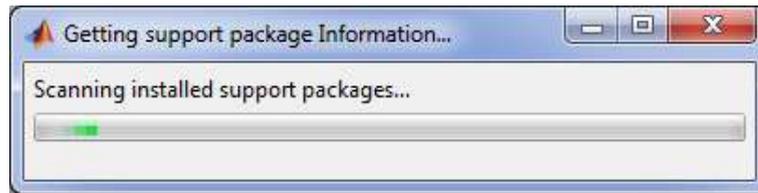
**Figura 1-C:** Instalación de paquetes adicionales.

Realizado por: JARA, Caroline; CABA, Carlos 2018

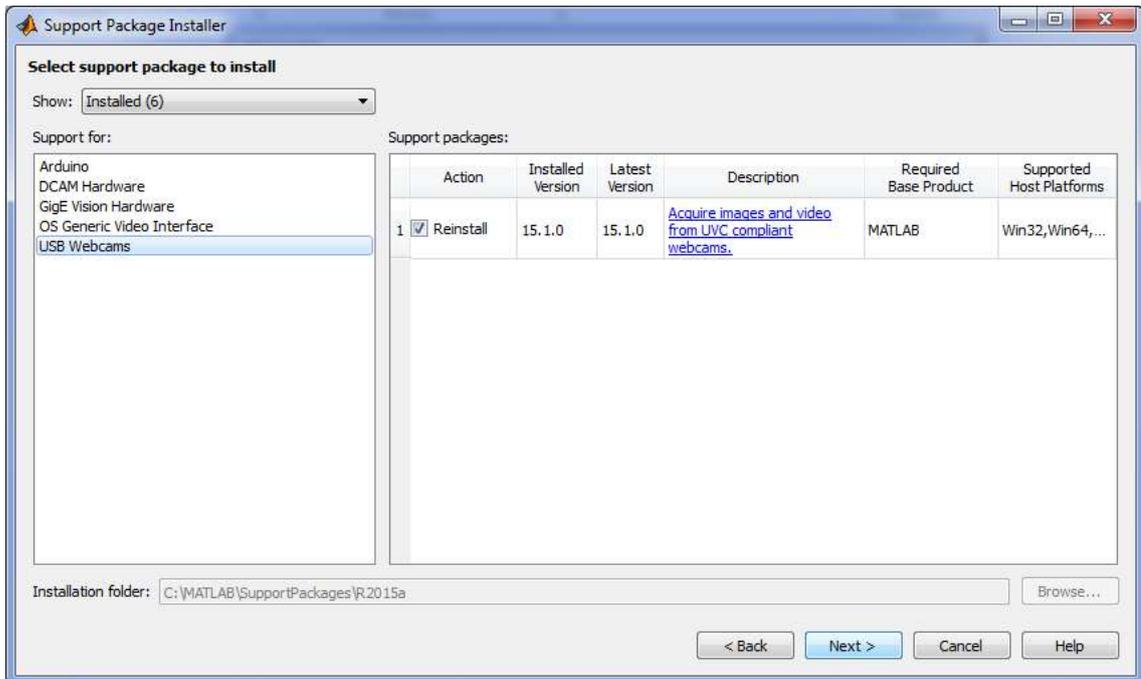


**Figura 2-C:** Instalación de paquetes adicionales.

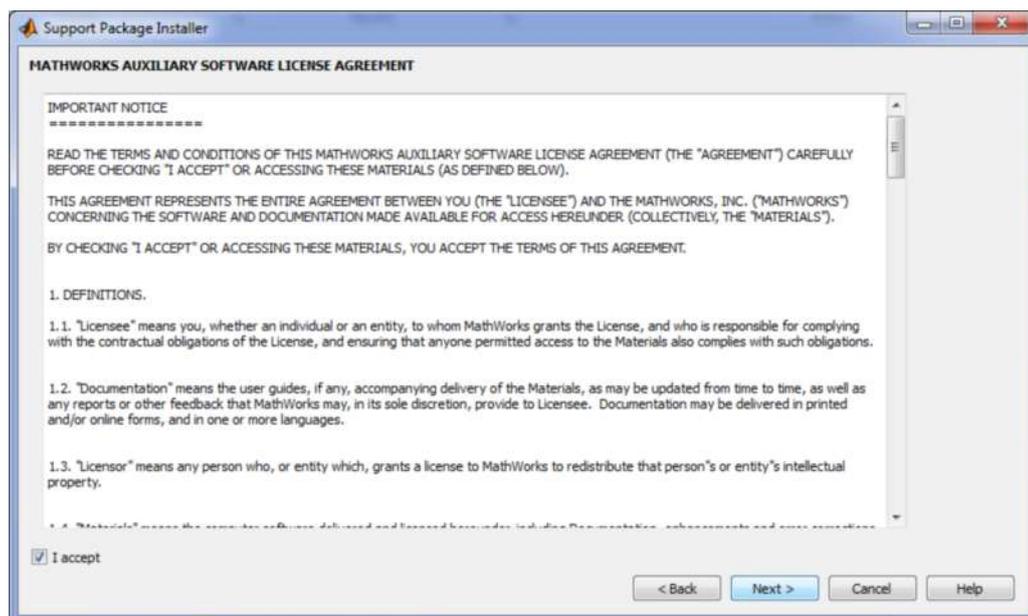
Realizado por: JARA, Caroline; CABA, Carlos 2018



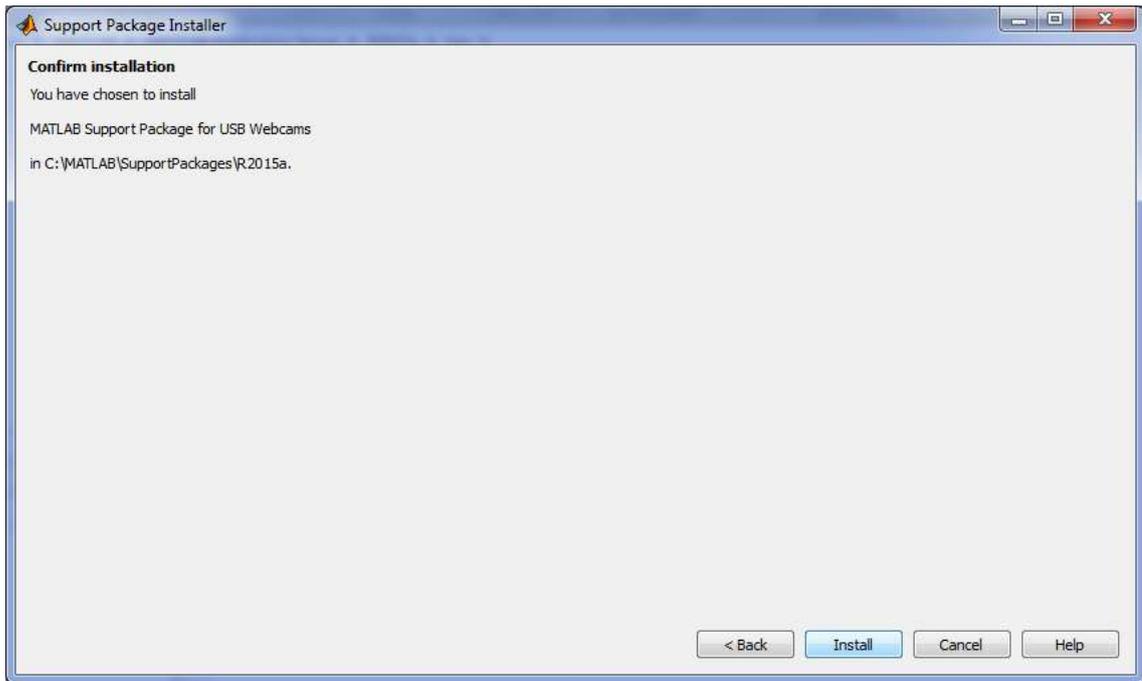
**Figura 3-C:** Instalación de paquetes adicionales.  
Realizado por: JARA, Caroline; CABA, Carlos 2018



**Figura 4-C:** Instalación de paquetes adicionales.  
Realizado por: JARA, Caroline; CABA, Carlos 2018

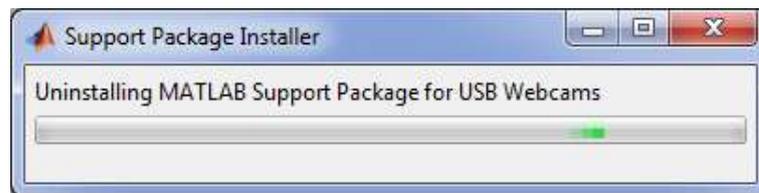


**Figura 5-C:** Instalación de paquetes adicionales.  
Realizado por: JARA, Caroline; CABA, Carlos 2018



**Figura 6-C:** Instalación de paquetes adicionales.

Realizado por: JARA, Caroline; CABA, Carlos 2018



**Figura 7-C:** Instalación de paquetes adicionales.

Realizado por: JARA, Caroline; CABA, Carlos 2018



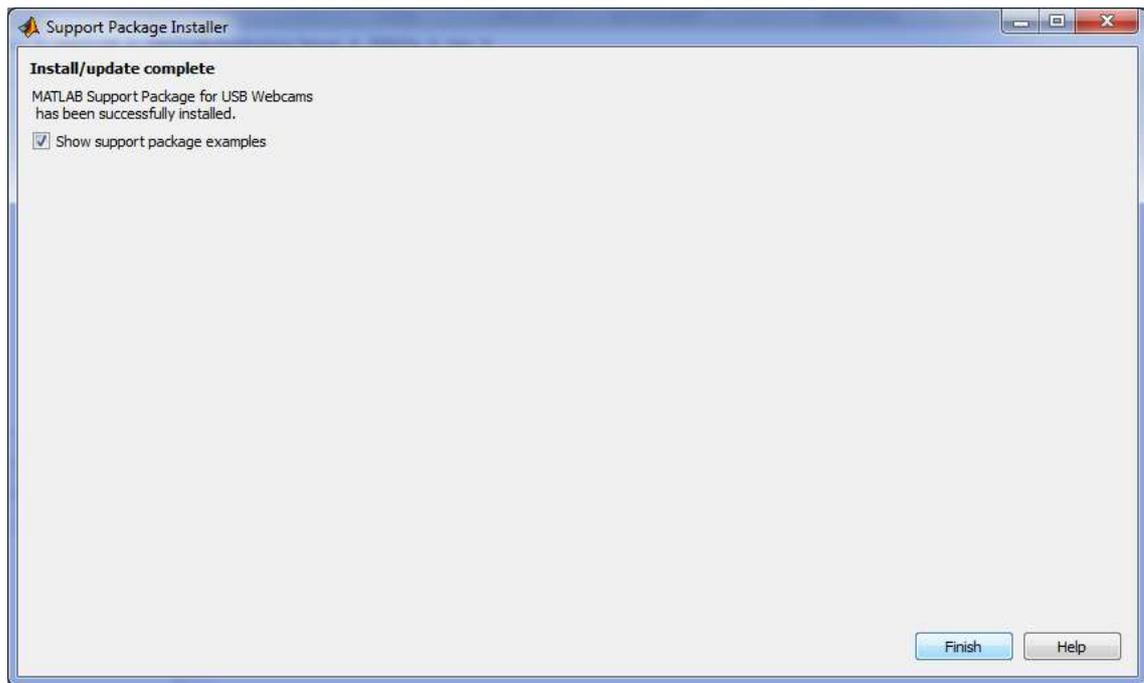
**Figura 8-C:** Instalación de paquetes adicionales.

Realizado por: JARA, Caroline; CABA, Carlos 2018



**Figura 9-C:** Instalación de paquetes adicionales.

Realizado por: JARA, Caroline; CABA, Carlos 2018



**Figura 10-C:** Instalación de paquetes adicionales.  
**Realizado por:** JARA, Caroline; CABA, Carlos

## ANEXO D

### MANUAL DE USUARIO.

#### MANUAL DE FUNCIONAMIENTO DEL SISTEMA.

##### REGISTRO DE USUARIO.

En la pantalla principal de nuestro sistema vemos que nos ofrece la opción de registro para lo cual cuenta con cuatro campos a llenar:

El campo nombre, contraseña y confirmar contraseña todos conocemos y son los básicos para un registro, pero adicional posee un campo “ADMIN” que un campo con el cual tendríamos que conocer la contraseña de administrador para poder registrar un nuevo usuario, en este caso la contraseña de administrador es “1234”.

The screenshot shows a web application interface with a light green background. At the top, a white box contains the text 'PROYECTO FINAL CAROLINE JARA - CARLOS CABA PRINCIPAL'. Below this, there are two main panels. The left panel, titled 'INGRESO', contains two input fields: 'NOMBRE:' and 'CONTRASEÑA:'. Below these fields is a button labeled 'ENTRAR'. The right panel, titled 'REGISTRO', contains four input fields: 'NOMBRE:' (with the value 'santiago'), 'CONTRASEÑA:' (with '\*\*\*\*'), 'CONFIRMAR CONTRASEÑA:' (with '\*\*\*\*'), and 'ADMIN:' (with '1234'). Below these fields is a button labeled 'REGISTRAR'. At the bottom center of the interface is a button labeled 'SALIR'.

**Figura 1-D:** Pantalla principal.

**Realizado por:** JARA, Caroline; CABA, Carlos 2018

Una vez que hayamos completado los campos correctamente y luego de dar clic en el botón registrar tendremos una ventana en donde se nos muestran las dos imágenes del usuario tanto de frente como de perfil, aquí en esta ventana viene una indicación donde el usuario debe permanecer en la forma precisa y cuando se aprecie bien dar clic en el botón capturar y esperar que se nos muestre los rostros, cuando el usuario este de acuerdo a las fotografías dar en continuar, caso contrario dar clic en repetir para repetir el proceso de captura.



**Figura 2-D:** Registro de usuarios.  
**Realizado por:** JARA, Caroline; CABA, Carlos 2018

Cuando el sistema haya detectado correctamente los rostros decidiremos si guardarlos o repetir la captura, si hemos decidido aceptar estas capturas daremos clic en continuar.



**Figura 3-C:** Registro de usuarios.  
**Realizado por:** JARA, Caroline; CABA, Carlos 2018

Una vez acabado el proceso de registro del usuario automáticamente nos enviará a la pantalla de visualización en donde nos dan varias opciones como:

- La opción de apertura de la puerta manual al dar clic en el botón “Permitir Ingreso”.
- La opción de ver detalles tanto de personas admitidas como de personas rechazadas.
- La opción de actualizar para visualizar algún cambio en la lista de admitidos o rechazados.

- La opción de regresar al menú principal para dar posible ingreso a otro usuario o registrar a uno nuevo.

Ahora bien si se ha decidido ver los detalles de algún usuario admitido se muestra la siguiente pantalla en donde podremos ver su nombre, porcentaje de similitud, y hora y fecha de ingreso, adicional de las dos fotografías y la opción de volver a visualización.



**Figura 4-D:** Detalles de usuarios admitidos.

Realizado por: JARA, Caroline; CABA, Carlos 2018

Si la opción elegida fue ver los detalles de un usuario rechazado lo que nos mostrará en pantalla es la imagen de frente, y la reconstrucción 3D, así como también la hora y fecha que el usuario ha sido rechazado y la opción para volver a la pantalla de visualización.

## INGRESO DE USUARIO.

Nuevamente en la pantalla principal tendremos disponibles los campos de nombre, contraseña para que un usuario ya registrado ingrese con sus datos haciendo clic en el botón “Entrar”

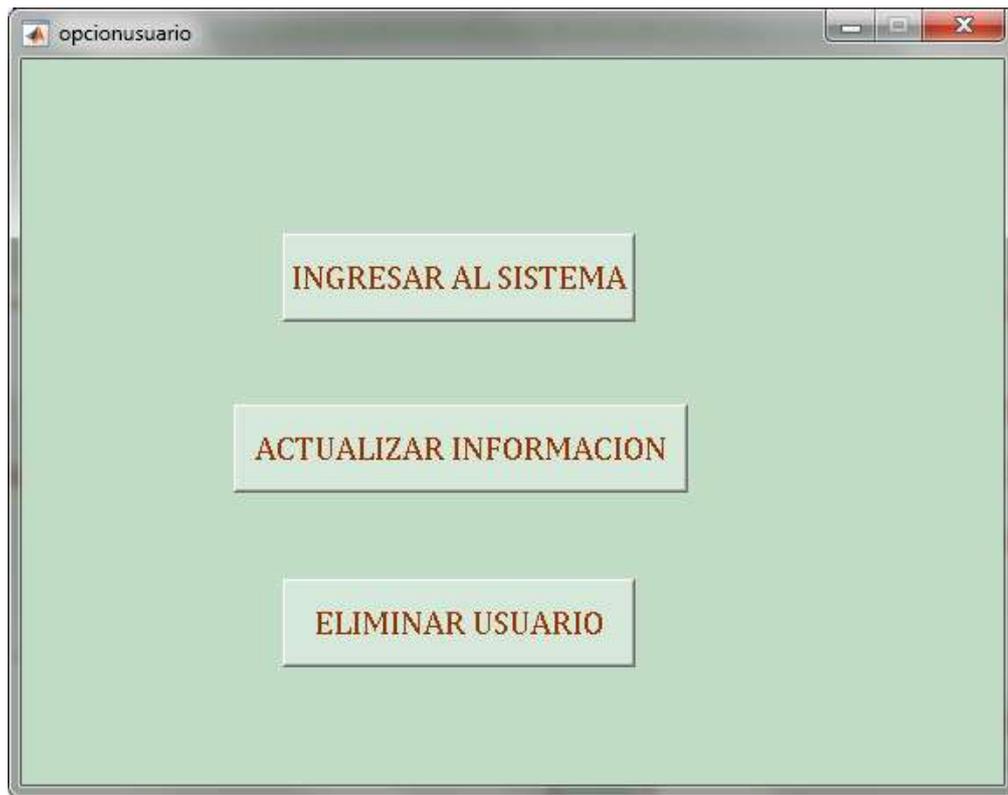


The screenshot shows a window titled 'principal' with a light green background. At the top center, a white box contains the text 'PROYECTO FINAL CAROLINE JARA - CARLOS CABA' and 'PRINCIPAL' below it. Below this, there are two main panels. The left panel is titled 'INGRESO' and contains a 'NOMBRE:' field with the text 'santiago', a 'CONTRASEÑA:' field with four asterisks, and an 'ENTRAR' button. The right panel is titled 'REGISTRO' and contains a 'NOMBRE:' field, a 'CONTRASEÑA:' field, a 'CONFIRMAR CONTRASEÑA:' field, and an 'ADMIN:' field, with a 'REGISTRAR' button below them. At the bottom center of the window is a 'SALIR' button.

**Figura 5-D:** Pantalla principal.

**Realizado por:** JARA, Caroline; CABA, Carlos 2018

Una vez que se ha comprobado que el usuario ingresado y la contraseña son las correctas se dará paso al usuario a tres opciones para elegir, ingresar al sistema donde se dará paso a la ventana de visualización, actualizar información sirve para cambiar ya sea el nombre de usuario su contraseña o sus fotografías, y eliminar el usuario para cuando decidan que ya no debe estar registrado ni tener acceso al sistema.



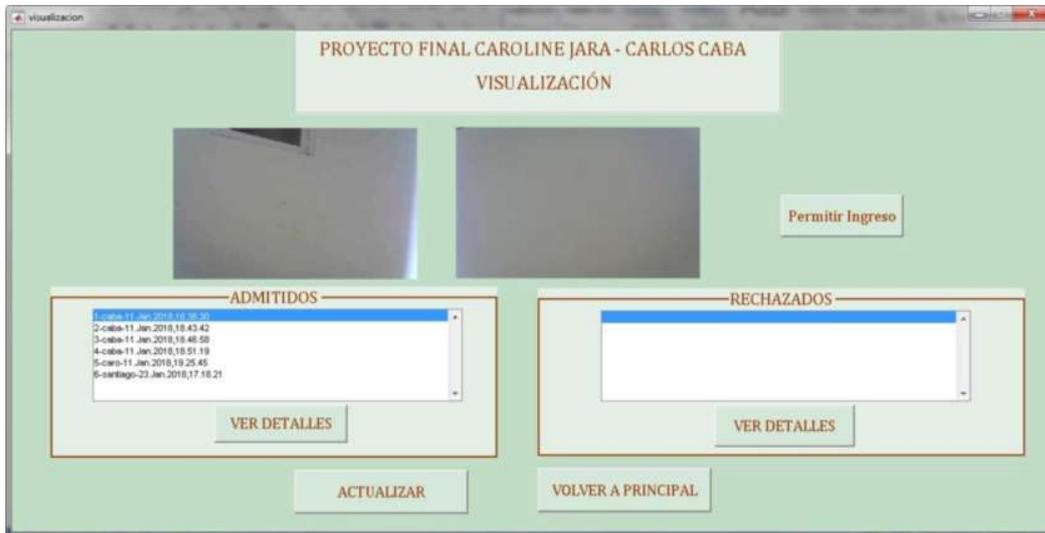
**Figura 6-D:** Opción de usuario.  
Realizado por: JARA, Caroline; CABA, Carlos 2018

Si la opción elegida fue actualizar información se pedirá la nueva información y se dará a elegir entre conservar las fotografías anteriores o tomar nuevas fotografías, dependiendo de la elección del usuario se enviará a la pantalla de toma de fotos o a la pantalla de visualización pero ya actualizado los datos.

A screenshot of a software window titled "nuevousuario". The window has a light gray background and contains a registration form titled "REGISTRO" in a green-bordered box. The form includes three text input fields: "NOMBRE:" with the value "santiago", "CONTRASEÑA:" with the value "caba", and "CONFIRMAR CONTRASEÑA:" with the value "caba". Below the fields are two radio buttons: "CONSERVAR IMAGENES" (selected) and "TOMAR NUEVAS IMAGENES". At the bottom of the form is a button labeled "ACTUALIZAR". The window title bar shows standard minimize, maximize, and close buttons.

**Figura 7-D:** Actualización de usuario.  
Realizado por: JARA, Caroline; CABA, Carlos 2018

Ahora bien si el usuario que intenta ingresar únicamente quiere paso a la pantalla de visualización se le concederá en donde tiene las opciones descritas en el apartado de registro de usuario.



**Figura 8-D:** Pantalla de visualización.  
Realizado por: JARA, Caroline; CABA, Carlos 2018