



**UNIVERSIDAD AUTÓNOMA DEL ESTADO DE MÉXICO**  
**CENTRO UNIVERSITARIO UAEM ATLACOMULCO**



“Clasificación de piezas metálicas mediante una Red Neuronal Convolutiva y  
Análisis de Componentes Principales”

# **T E S I S**

Que para obtener el Grado Académico de:

**Maestra en Ciencias de la Computación**

Presenta:

**Mayra Mateo Jiménez**

Director de Tesis:

**Dr. Carlos Eduardo Torres Reyes**

Co-directores de tesis:

**Dr. Everardo Efrén Granda Gutiérrez**

**Dr. José Arturo Pérez Martínez**

**Mes del Año**

## **AGRADECIMIENTOS**

Agradezco al Consejo Nacional de Ciencia y Tecnología (CONACyT) por la beca recibida durante la Maestría en Ciencias de la Computación, misma que sirvió para el desarrollo de la investigación y publicaciones realizadas.

Expreso mi agradecimiento al tutor de tesis de este proyecto, esencialmente por su profesionalismo, tiempo, consejos, orientación e instrucción para el desarrollo de la investigación.

Así mismo, externo el agradecimiento para el comité tutorial; por su tiempo, dedicación y valiosos aportes para que esta investigación fuera posible.

Agradezco a mis padres por su apoyo y consejos para culminar los estudios de la maestría.

## **PUBLICACIONES**

Mateo-Jiménez, M., Granda-Gutiérrez, E.E., Rangel-Velázquez G.I., Torres-Reyes, C.E. y Pérez-Martínez, J.A., 2021. Implementación de técnicas de visión artificial en un sistema de reconocimiento facial para alertar el robo de automóviles. Journal Aristas: Investigación básica y aplicada, Vol. 8 , N°16. pp. 219-225. ISSN: 2007-9478.

### **Abstract**

Facial recognition is used in applications for attendance registration, people identification or security systems. In the last, its purpose is to detect people, whether intruders or allowed users in the system, through the collection of biometric data; for example: the distance of the eyes, mouth, or nose. These data are unique, allowing to differentiate a person from others by using algorithms for image analysis. In this work, three algorithms are used: Eigenfaces, Principal Component Analysis and Local Binary Pattern Histogram, to perform facial recognition inside a car, since these algorithms are considered of low computational cost. The proper designed tool sends e-mail alerts and a link to a photograph, as well as a message to the owner's cell phone in case the person enters is an intruder; otherwise, the owner can deactivate the system or decide whether to proceed with the facial recognition. Each alert is generated every 5 minutes, to allow the GPS location to be updated. As the correct classification is a relevant parameter, the accuracy metric is evaluated as 97 %, with respect to the confusion matrix.

Mateo-Jiménez, M., Torres-Reyes, C.E., Granda-Gutiérrez, E.E. y Pérez-Martínez, J.A., 2021. Analysis and comparison of a classification of regular metal pieces by Convolutional Neural Networks with and without Principal Component Analysis. *Journal Research in Computing Science (CORE 2021)*, ISSN: 1870-4069, México. Aceptado.

### **Abstract**

The classification of images using artificial intelligence techniques (such as neural networks) is fundamentally related to the characteristics of images, such as having a background of a single uniform color. If the illumination scale of the images is different, or the resolution of the image is not sufficient, this may cause false positives when classifying. This work analyzes the importance of classifying images of metal parts (with regular shapes, such as circle, square, rectangle, and flat washer, using a convolutional neural network with and without principal component analysis (PCA), to determine the percentage of classification of images in comparison with the two algorithms by evaluating sensitivity, specificity, accuracy, and precision. The methodology is divided into three steps: obtaining the dataset, training the convolutional neural network, and validating its results. Some representative results of a comparison with a convolutional neural network without PCA were obtained using the RMSProp optimizer with 85.3 % precision. PCA achieved the accuracy 98.7 %, which indicates that the implementation of PCA improved the classification, with 480 components used.

Rangel-Velázquez, G.I., Ambriz-Polo, J.C., Mateo-Jiménez, M., Pérez-Martínez, J.A., Granda-Gutiérrez, E.E. y Torres-Reyes, C.E., 2021. Sistema de digitalización de texto manuscrito implementando Google Vision API. *Journal Aristas: Investigación básica y aplicada*, Vol. 8 , N°16. pp. 325-331. ISSN: 2007-9478

### **Abstract**

Image processing, together with pattern recognition, have been areas of study exploited in recent years, achieving advances such as object classification, face recognition and text recognition. In particular, the latter allows handwritten, typed, or printed texts to be converted into editable digital text. This paper describes the design and operation of a handwritten text digitization system through the implementation of Google Vision API, oriented for Android devices. The aim of this research is to verify if its use increases the efficiency in recognizing autographic text due to the low performance of Optical Character Recognition (OCR) systems when processing text of such a type. The developed system consists of three modules: 1) Image acquisition, 2) API consumption request and 3) Digitization of the generated OCR. For the evaluation of its performance, eleven document formats corresponding to the education, health and industry areas were used, and four different image conditions (quality adjustment and cropping of the area of interest), as well as a comparison with some of the existing applications on the market. In this sense, the average recognition of handwritten words with respect to those contained in each format was calculated and a 67 % efficiency of the system was determined.

## RESUMEN

Las redes neuronales artificiales tienen diversas aplicaciones tales como predicción de sucesos, procesamiento de datos o automatización de actividades en detección y clasificación de objetos. Un ejemplo de los entornos en que se emplean las redes neuronales artificiales es la manufactura, donde algunas actividades se realizan de manera manual, como colocar la herramienta de trabajo sobre la pieza de trabajo, conllevando situaciones en donde se involucra el error humano en el desarrollo de los procesos que es causado por el cansancio. Por lo tanto, una alternativa de solución es la automatización a través de la detección de objetos. Por otra parte, al utilizar sistemas de visión artificial, intervienen variables que tienen impacto en la percepción visual como principalmente la tonalidad de la pieza, tipos de sensores de la cámara, ángulo de la posición de la pieza respecto a la cámara y los cambios de iluminación, estos últimos son más evidentes en piezas metálicas por reflejar la luz. En el presente trabajo se muestra el desarrollo de una Red Neuronal Convolutiva (RNC) con Análisis de Componentes Principales (PCA) y la Red Neuronal Convolutiva sin el Análisis de Componentes Principales para realizar la clasificación de piezas metálicas de acuerdo con su forma geométrica, ya sea regular e irregular. Para la experimentación se utiliza el diseño factorial  $3^4$ , con las variables: iluminación, cantidad de neuronas en capa oculta, tipo de optimizador y número de componentes, que permitan obtener al menos el 93% de exactitud. Las piezas metálicas son expuestas a luz fría y cálida a tres niveles de iluminación: 1000 lx, 1500 lx y 2000 lx, establecidos en la Norma Oficial Mexicana NOM-025-STPS-2008. A través del pretratamiento de imágenes con Análisis de Componentes Principales se elimina el ruido que propician los datos redundantes (debido usualmente a la dimensión de los datos) y se obtiene una dimensión menor a la original, esto permite que la Red Neuronal Convolutiva reciba una cantidad menor de datos de entrada. Las etapas del presente trabajo se desglosan en: adquisición del *dataset* con cámaras de 1.3 MP y 8 MP, el diseño de experimentos que contempla las variables antes definidas para la Red Neuronal Convolutiva y la Red Neuronal Convolutiva con algoritmo Análisis de Componentes Principales para el pretratamiento de imágenes, entrenamiento y validación. En la última

etapa se presentan los resultados obtenidos con la Red Neuronal Convolutiva se obtuvo un 78.89 % de exactitud, 76.66 % de sensibilidad, 79.85 % de precisión y F1 (Puntuación F1) con 76.27 % y con Análisis de Componentes Principales se obtiene un promedio del 95.75 % de exactitud, 95.22 % de sensibilidad, 95.80 % de precisión y 95.50 % de F1. Estos últimos resultados demuestran que el uso del Análisis de Componentes Principales en la Red Neuronal Convolutiva mejora el reconocimiento de imágenes de piezas metálicas a través de la eliminación de información redundante en las imágenes, obteniendo una desviación estándar del 2.4 %, lo cual asegura que los datos de clasificación se aproximan al 95 % de clasificación.

**Palabras clave: Análisis de Componentes Principales, Red Neuronal Convolutiva, iluminación, clasificación, piezas metálicas.**

## **ABSTRACT**

Artificial neural networks have diverse applications such as event prediction, data processing or automation of activities in object detection and classification. An example of the environments in which artificial neural networks are used is manufacturing, where some activities are performed manually, such as placing the work tool on the workpiece, leading to situations where human error is involved in the development of processes that are caused by fatigue. Therefore, an alternative solution is automation through object detection. On the other hand, when using artificial vision systems, variables that have an impact on visual perception intervene, such as mainly the tonality of the part, types of camera sensors, the angle of the part position with respect to the camera and lighting changes, the latter being more evident in metallic parts due to light reflection. This work shows the development of a Convolutional Neural Network (CNN) with Principal Component Analysis (PCA) and the Convolutional Neural Network without Principal Component Analysis to perform the classification of metallic parts according to their geometric shape, either regular or irregular. For the experimentation, the 34 factorial design is used, with the variables: illumination, number of neurons in hidden layer, type of optimizer and number of components, which allow obtaining at least 93% of accuracy. The metal parts are exposed to cold and warm light at three illumination levels: 1000 lx, 1500 lx and 2000 lx, established in the Mexican Official Standard NOM-025-STPS-2008. Through the pretreatment of images with Principal Component Analysis, the noise caused by redundant data (usually due to the dimension of the data) is eliminated and a dimension smaller than the original one is obtained, this allows the Convolutional Neural Network to receive a smaller amount of input data. The stages of the present work are broken down into: acquisition of the dataset with 1.3 MP and 8 MP cameras, the design of experiments that contemplates the variables previously defined for the Convolutional Neural Network and the Convolutional Neural Network with Principal Component Analysis algorithm for image pretreatment, training and validation. In the last stage, the results obtained with the Convolutional Neural Network were 78.89 % of accuracy, 76.66 % of sensitivity, 79.85 % of precision and F1 (F1 Score) with 76.27 % and with



Principal Component Analysis an average of 95.75 % of accuracy, 95.22 % of sensitivity, 95.80 % of precision and 95.50 % of F1 is obtained. These last results demonstrate that the use of Principal Component Analysis in the Convolutional Neural Network improves the recognition of images of metal parts through the elimination of redundant information in the images, obtaining a standard deviation of 2.4 %, which ensures that the classification data is close to 95 % classification.

**Keywords: Principal Component Analysis, Convolutional Neural Network, illumination, classification, metal parts.**

# ÍNDICE

AGRADECIMIENTOS.....	2
PUBLICACIONES.....	3
RESUMEN.....	6
ABSTRACT.....	8
ÍNDICE.....	10
ÍNDICE DE TABLAS.....	13
ÍNDICE DE FIGURAS.....	14
1 INTRODUCCIÓN.....	18
2 PLANTEAMIENTO DEL PROBLEMA.....	26
2.1 Definición del problema.....	26
2.2 Objetivos de investigación.....	27
2.3 Preguntas de investigación.....	28
2.4 Justificación.....	28
2.5 Impactos.....	29
3 HIPÓTESIS.....	30
4 ESTADO DEL ARTE.....	31
4.1 Imagen.....	32
4.2 Factores que intervienen en Visión Artificial.....	37
4.2.1 Guiado.....	37
4.2.2 Inspección.....	38
4.2.3 Iluminación.....	40
4.3 Clasificador: Red Neuronal Artificial (RNA).....	41

4.3.1	Red Neuronal Convolutacional .....	43
4.4	Análisis de Componentes Principales .....	52
4.5	Métricas de evaluación. ....	56
5	MÉTODO.....	61
5.1	Diseño factorial $3^4$ .....	61
5.2	Proceso general del algoritmo .....	66
5.3	Requerimientos o especificaciones del sistema .....	68
5.4	Diseño e implementación.....	70
5.4.1	Selección de piezas metálicas .....	70
5.4.2	Establecimiento de las condiciones de iluminación .....	70
5.4.3	Captura de imágenes.....	72
5.4.4	Tratamiento de las imágenes .....	75
5.4.5	Diseño y codificación de la red neuronal .....	76
5.4.6	Etapa de entrenamiento de la red neuronal.....	77
5.4.7	Etapa de validación de Red Neuronal Convolutacional .....	79
5.4.8	Desarrollo de interfaz para el manejo del sistema .....	81
6	RESULTADOS Y DISCUSIÓN.....	87
6.1	Normalización de imágenes .....	87
6.2	Pretratamiento de imágenes con PCA.....	93
6.3	Resultados del diseño factorial $3^4$ .....	102
6.4	Evaluación de los resultados de entrenamiento con y sin PCA .....	106
6.5	Validación .....	112
6.6	Interfaz para el manejo del sistema.....	115

7	CONCLUSIONES .....	124
8	REFERENCIAS .....	127
9	ANEXOS.....	135

## ÍNDICE DE TABLAS

<i>Tabla 1. Operaciones morfológicas</i> .....	24
Tabla 2. Comparación de los tipos de arquitectura de redes neuronales convencionales. ...	48
Tabla 3. Comparación de los tipos de arquitectura de redes neuronales de aprendizaje profundo.....	49
Tabla 4. Principales funciones de activación para el análisis de datos en Redes Neuronales. ....	51
Tabla 5. Recopilación de trabajos que usaron técnicas de Visión Artificial. ....	58
Tabla 6. Matriz de experimentación. ....	66
Tabla 7. Ejemplo de combinaciones de matriz de experimentos .....	66
Tabla 8. Características de las cámaras de 1.3 MP y 8MP.....	69
Tabla 9. Imágenes por pieza para entrenamiento .....	74
Tabla 10. Imágenes por pieza para validación .....	74
Tabla 11. Desviación estándar de la exactitud obtenida por cada componente. ....	101
Tabla 12. Resultado de promedios en las métricas durante el entrenamiento .....	111
Tabla 13. Intervalos de confianza, con un tamaño de muestra de 500, en la validación....	115
Tabla 14. Resultados de la prueba 1 del diseño factorial $3^4$ . ....	135
Tabla 15. Resultados de la prueba 1 del diseño factorial $3^4$ (2). ....	136
Tabla 16. Resultados de la prueba 1 del diseño factorial $3^4$ (3) .....	137
Tabla 17. Resultados de prueba 2 del diseño factorial $3^4$ .....	137
Tabla 18. Resultados de la prueba 2 del diseño factorial $3^4$ (2) .....	137
Tabla 19. Resultados de la prueba 2 del diseño factorial $3^4$ (3) .....	139
Tabla 20. Resultados de la prueba 3 del diseño factorial $3^4$ .....	140
Tabla 21. Resultados de la prueba 3 del diseño factorial $3^4$ (2) .....	140
Tabla 22. Resultados de la prueba 3 del diseño factorial $3^4$ (3) .....	141
Tabla 23. Hoja de datos de luxómetro UT382 Series. ....	142

## ÍNDICE DE FIGURAS

Figura 1. Procedimiento para reducir matriz original.....	21
Figura 2. Resultado de reducir matriz original con PCA. ....	22
Figura 6. Matriz de NxM dimensiones .....	33
Figura 7. Ejemplo de muestreo.....	33
Figura 8. Características de la imagen.....	34
Figura 9. Imagen con brillo; el recuadro en naranja ocasiona problemas en la identificación. .....	35
Figura 10. Tipos de iluminación.....	36
Figura 11. Codo de 90°.....	38
Figura 12. Identificación de una pieza en distinta posición. ....	38
Figura 13. Identificación de irregularidades.....	39
Figura 14. Componentes de un sistema de Visión Artificial, adaptado de (Cognex, 2020). 39	
Figura 15. Modelo de una Neurona Artificial, adaptado de (Salazar Cárdenas y Vargas Trujillo, 2017).....	42
Figura 16. Etapas de la Red Neuronal Convolutiva, adaptado de (Scherer, Müller, y Behnke, 2010).....	45
Figura 17. Aplicación de filtro, (Khan, Rahmani, Shah, y Bennamoun, 2018) .....	45
Figura 18. Ejemplo de histograma.....	54
Figura 19. Proceso de Análisis de Componentes Principales.....	55
Figura 20. Matriz de confusión.....	58
Figura 21. Etapas de experimentación.....	61
Figura 22. Etapas del diseño factorial .....	62
Figura 23. Relación entre factores.....	63
Figura 24. Comportamiento de la varianza acumulada en función del número de componentes .....	64
Figura 25. Gráficas del promedio de exactitud para los optimizadores SGD, Adam, RMSProp, Nadam y Adadelta .....	65

Figura 26. Etapas del proceso de PCA en RNC .....	68
Figura 27. Piezas metálicas regulares .....	70
Figura 28. Piezas metálicas irregulares .....	70
Figura 29. Diagrama electrico de conexión dimmer-lámpara .....	71
Figura 30. Entorno de trabajo. ....	71
Figura 31. Algoritmo para captura y recorte de imágenes. ....	73
Figura 32. Balance de iluminación .....	75
Figura 33. Pasos para el desarrollo del proceso de Análisis de Componentes Principales. .	75
Figura 34. Red Neuronal Convolutacional .....	77
Figura 35. Proceso de entrenamiento de la Red Neuronal Convolutacional con y sin PCA...	79
Figura 36. Proceso para matriz de confusión .....	80
Figura 37. Diagrama entidad relación de la interfaz del sistema.....	81
Figura 38. Diagrama de maquetación de inicio .....	82
Figura 39. Diagrama de maquetación de Registro de nuevo usuario .....	82
Figura 40. Diagrama de maquetación de Acceso al sistema. ....	83
Figura 41. Diagrama de maquetación de funciones para el sistema.....	83
Figura 42. Diagrama de maquetación de los resultados clasificación con matriz de confusión. .....	84
Figura 43. Diagrama de los resultados de clasificación de imágenes contenidas en carpeta. .....	84
Figura 44. Diagrama de secuencias del sistema de clasificación de piezas metálicas. ....	86
Figura 45. Diseño del entorno de trabajo real. Los elementos de la imagen son: 1.-Mesa de trabajo, 2.- Lámparas de 100W, 3.- Cámara, 4.- Fondo negro, 5.- Luxómetro UT380 Series, 6.- Dimmer, 7.- Pieza de metal, 8.- Computadora donde se aloja el sistema y se conecta la cámara USB.....	87
Figura 46. Proceso de redimensionamiento.....	88
Figura 47. Resultados del balance de iluminación con la imagen del círculo a 1500 lx .....	89
Figura 48. Muestra de balance de iluminación con luz cálida en imágenes de 1.3 MP en cada nivel de iluminación (1).....	90

Figura 49. Muestra de balance de iluminación con luz cálida en imágenes de 1.3 MP en cada nivel de iluminación (2).....	91
Figura 50. Balance de iluminación en imágenes de 1.3 MP con iluminación fría. ....	92
Figura 51. Balance de iluminación en imágenes de 8 MP a) iluminación cálida, b) iluminación fría.....	92
Figura 52. Varianza acumulada de los componentes PCA.....	94
Figura 53. Iteraciones con 2 componentes. ....	95
Figura 54. Entrenamiento con 4 componentes. ....	97
Figura 55. Entrenamiento con 10 componentes. ....	98
Figura 56. Entrenamiento con 25 componentes. ....	98
Figura 57. Entrenamiento con 37 componentes. ....	99
Figura 58. Entrenamiento con 50 componentes. ....	100
Figura 59. Promedio de las métricas de 2, 4, 10, 25, 37 y 50 componentes.....	101
Figura 60. Proyección de componentes.....	102
Figura 61. Gráficas del promedio de exactitud, con 1000 lx. a) imágenes de 1.3 MP. b) imágenes con 8 MP.....	103
Figura 62. Gráficas del promedio de exactitud, con 1500 lx. a) imágenes de 1.3 MP. b) imágenes con 8 MP.....	104
Figura 63. Gráficas del promedio de exactitud, con 2000 lx. a) imágenes de 1.3 MP. b) imágenes con 8 MP.....	105
Figura 64. Resultados con optimizador a) Adam y b) RMSProp.....	106
Figura 65. Entrenamiento y pérdida, con y sin PCA .....	107
Figura 66. Resultados de entrenamiento y pérdida sin PCA: a) y c). con PCA: b) y d) ....	108
Figura 67. Resultados de entrenamiento. Sin PCA a) y con PCA b) .....	110
Figura 68. Predicciones realizadas. ....	112
Figura 69. Prueba de predicciones con balance de iluminación. ....	112
Figura 70. Matriz de confusión sin PCA con imágenes de 1.3 MP.....	113
Figura 71. Ejemplo de matriz de confusión con imágenes de 1.3 MP. ....	114
Figura 72. Archivos del sistema. ....	116



Figura 73. Menú de inicio.....	116
Figura 74. Registro de usuario.....	117
Figura 75. Registro existente.....	117
Figura 76. Registro de nuevo usuario.....	118
Figura 77. Almacenamiento de datos de registro de usuario.....	118
Figura 78. Entrar como usuario.....	119
Figura 79. Mensaje de registro no encontrado.....	119
Figura 80. Menú del sistema.....	120
Figura 81. Clasificación de clavo.....	121
Figura 82. Clasificación de rectángulo.....	121
Figura 83. Clasificación de imágenes en carpeta.....	121
Figura 84. Resultados de clasificación (2).....	122
Figura 85. Aviso de espera para generación del resultado estadístico.....	122
Figura 86. Matriz de confusión con resultados de métricas.....	123

# 1 INTRODUCCIÓN

En el ámbito de la manufactura de piezas metálicas se emplean actividades como reconocimiento para determinar: el tamaño, área y forma de la pieza que se está maquinando. Para lograrlo, es conveniente utilizar un sistema que sea preciso y capaz de reconocer el objeto con condiciones específicas de iluminación (Younas, Kumari, y Manzoor, 2017).

El producto resultante del maquinado es utilizado en equipos electrónicos, médicos, de transporte como terrestres, aéreos, entre otros, por lo que se exige que las piezas tengan un mínimo error de maquinado, siendo éste relevante en aquellas de menor tamaño, porque en algunas ocasiones el proceso realizado es medido en el orden de las micras para tener un funcionamiento adecuado y acorde a las características requeridas (Utrilla Diaz y Reyes Hernández, 2019).

Para que a las piezas metálicas se les dé una forma determinada, en ocasiones es necesario usar algunas técnicas de maquinado no convencionales, donde no se tiene el contacto directo entre la herramienta de trabajo y la pieza, como puede ser el proceso de maquinado electroquímico, maquinado por chorro de agua, maquinado ultrasónico o como en el caso del maquinado por descarga eléctrica que implica el uso de una herramienta de trabajo, pieza de trabajo, una determinada concentración de la solución electrolítica como cloruro de sodio y nitrato de sodio comúnmente usadas y un determinado voltaje de corriente directa o por corriente pulsada, menor a 20 V (Chaides, Ahuett, Flores, Caballero, y Ruíz, 2009).

También existen las técnicas convencionales como el barrenado, torneado, fresado y prensado, que involucran ciertas condiciones como el contacto directo con la pieza que genera fricción, el aumento de temperatura y velocidad de avance durante el corte, obteniendo como efecto la disminución de la vida a fatiga del material (Díaz del Castillo Rodríguez, 2011) y (Petatan Bahena, Gutiérrez Rojas, Abúndez Pliego, Mayén Chaires, y Andrés, 2021).

En la operatividad del maquinado convencional o tradicional, se realizan actividades como la remoción de material mediante una herramienta de corte para dar forma a la pieza,

siguiendo lineamientos de los cuales uno de ellos es la constante intervención del operador para nuevamente colocar la herramienta de trabajo sobre la pieza de materia prima y prepararlas para realizar el maquinado. Como efecto de la participación humana en las actividades anteriores se produce el “cansancio” en el operador, que incrementa la probabilidad de generar error humano. Por otra parte, en un entorno de clasificación automatizado, se involucra la iluminación porque las variaciones afectan el impacto visual reflejado en la obtención de imágenes mediante la cámara, al clasificarlas con los algoritmos de reconocimiento de patrones, como las redes neuronales artificiales y al evaluar con diversas métricas que miden el desempeño como sensibilidad, precisión, exactitud y F1, puede diferir al valor considerado como adecuado, por ejemplo para el caso de la métrica de exactitud el mínimo es de 90% (Guillermo González, 2014).

Una posible solución a lo anterior es emplear un entorno controlado, mediante el uso de técnicas como iluminación frontal, direccional y estructurada para obtener mejor nitidez en las imágenes, en donde el sistema se encuentra restringido en calibrar correctamente esas variables, por lo tanto, es más factible emplear un sistema que sea adaptativo a las variaciones de iluminación para que pueda ser reproducible en otros entornos. (Sánchez Reinoso, 2011)

Para diferenciar un objeto de otro con visión artificial se pueden detectar algunos patrones como: forma geométrica, tamaño, color, profundidad, opacidad, rotación, posición, condiciones de iluminación y reflectancia. (González Marcos, Martínez de Pisón Ascacibar y Verónica Pernía, 2006).

Los algoritmos de visión artificial son sensibles en la distinción de la geometría de la pieza metálica porque puede provocar un error dentro del reconocimiento del objeto. Entonces se pueden clasificar como elementos nuevos, aunque no lo sean. Este tipo de error es conocido como falso negativo según las métricas de evaluación del rendimiento de sistemas de clasificación (Ariza Lopez, Rodríguez Avi, y Alba Fernández, 2018).

No obstante, para evitar que ocurran falsos negativos en la clasificación, se utilizan algunas técnicas como el uso de una Red Neuronal Artificial, siendo una herramienta que mejora los

datos de entrada, es decir, que las imágenes se encuentren normalizadas aplicando un filtro y una escala de redimensionamiento. Para normalizar los niveles de iluminación de la imagen es necesario considerar algunas características como el contraste y el efecto visual que ocasiona la incidencia de la luz sobre la superficie del objeto.

La mayoría de los procesos de reconocimiento de patrones utilizan condiciones de iluminación para que funcionen de manera adecuada en el entorno de trabajo, de lo contrario se producen falsos positivos o negativos, sin embargo, para solventar la problemática de las condiciones de iluminación se utilizan algoritmos para el tratamiento de imágenes como las Redes Neuronales Artificiales y el Análisis de Componentes Principales, con el fin de mejorar las características de las imágenes (Soares y Souza, 2016).

Un ejemplo del reconocimiento de objetos es el trabajo realizado por Morejón Ortega (Morejón Ortega, 2019) mediante el control de un brazo robótico, con una Red Neuronal Artificial, usando también algoritmos para la detección y representación del objeto como: Función de Objeto de Frontera (BOF) y Momentos Invariantes (MI). El uso de una RNA *backpropagation* con entrenamiento de Momentos Invariante obtuvo un 90.69 % de precisión, mientras que con la Función de Objeto de Frontera corresponde al 72.09 % de precisión, presentando un mejor rendimiento con la aplicación de RNA *backpropagation* implementada con Momentos Invariantes. Primero se procesaron los datos para extraer las características de los objetos y de esta manera representarlos con Momentos Invariantes, teniendo en cuenta la posición y rotación de la pieza, además de la forma geométrica de las piezas, lo que minimiza el error de clasificación y de pérdida de datos.

Además de los algoritmos utilizados por Morejón Ortega (Morejón Ortega, 2019), la literatura señala que las Redes Neuronales Artificiales tienen auge en el campo de la visión artificial, abordando soluciones de problemas como determinar diagnósticos médicos, análisis del comportamiento humano, detección de objetos, clasificación de objetos, entre otras aplicaciones; sin embargo, la cantidad de datos de entrada y la aplicación en donde se

utilicen las redes neuronales artificiales, las hace totalmente diferentes entre ellas (Aziz, Salam, Bin Haji Salam, Ullah Sheikh, y Ayub, 2020).

Los algoritmos para entrenar una red neuronal son dos, el primero es Función de Objeto de Frontera (BOF), utilizado para delimitar las áreas a identificar del objeto y el segundo es Momentos Invariantes (MI) que se utiliza para extraer las características de los objetos en las imágenes (Gitesh Chitaliya y Trivedi, 2010).

Por otro lado, se puede utilizar el Análisis de Componentes Principales (PCA) para reducir la cantidad de datos de entrada en el procesamiento de la red neuronal artificial, el procedimiento para realizar dicha reducción se presenta en la figura 1. Primero se obtiene la matriz original, después se identifica la cantidad de píxeles y de componentes que contenga la matriz, para luego determinar la correlación entre píxeles y descartar los que tengan menor cantidad. Después de eliminar los píxeles con correlación más baja se genera una nueva matriz y se evalúa el centro para obtener la varianza acumulada. La cantidad de varianza acumulada se evalúa a partir de los componentes necesarios para obtener una matriz reducida la cual contendrá la información más importante.

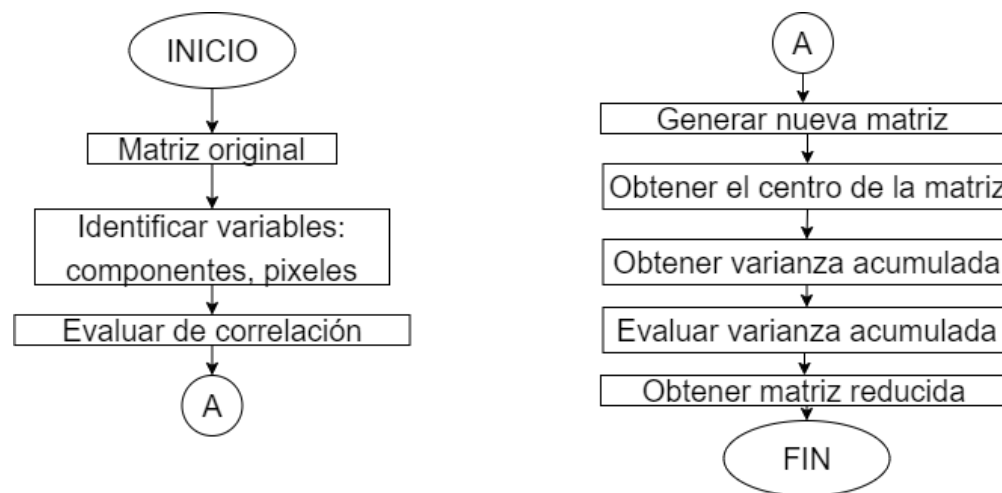
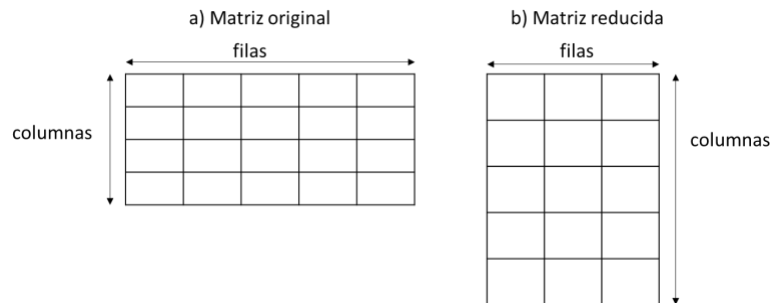


Figura 1. Procedimiento para reducir matriz original.

En la figura 2, se muestra un ejemplo de la reducción de datos utilizando el algoritmo de PCA. Las imágenes se presentan en forma de matrices con  $m$  filas y  $n$  columnas (Figura 2,

a), entonces al aplicar el algoritmo PCA, se puede reducir la dimensión de la matriz original (Figura 2, b) a través de la discriminación de información, después se obtiene el centro de la matriz para luego identificar las componentes principales.



*Figura 2. Resultado de reducir matriz original con PCA.*

Los pasos que se siguen para aplicar el PCA son los siguientes de acuerdo con Mazloom y Kasaei, (2005) (Mazloom y Kasaei, 2005):

1. Eliminar la correlación entre variables. Evaluar las variables que contenga menor cantidad de información y convertirlas en independientes, las cuales contienen la mayor información necesaria.
2. Determinar el centro de la matriz. Es importante conocer el centro de la matriz de los pixeles que se obtiene del promedio del conjunto de pixeles que resulte para  $m$  y del promedio que resulte para  $n$ .
3. Obtener varianza con PCA. La varianza representa la distancia de cada pixel al centro de la matriz. Para cada uno de los valores de varianza corresponde un componente, que indica la cantidad de unidades que constituyen un porcentaje de varianza. Por ejemplo, el componente 1 puede tener un 20 % de varianza, el segundo componente un 25 %.
4. Determinar cantidad de componentes. La cantidad de componentes depende de la matriz original y del tamaño al que se desea reducir la matriz. A cada uno de los valores obtenidos se le conoce como varianza individual e indica la cantidad de información por cada unidad de componente. La cantidad de componentes y varianza

adecuados dependen del estudio que se esté realizando, para este trabajo se evalúa la cantidad de componentes acumulados necesarios, por lo tanto, se utiliza la varianza acumulada que consiste en la suma del total de varianzas individuales obtenidas, por ejemplo, si se utilizaran 2 componentes con varianza acumulada, de acuerdo con el ejemplo anterior, se suma 20 % de la primer varianza y 25 % de la segunda, obteniendo como resultado el 45 % de varianza acumulada

5. Evaluar la varianza. Para cada uno de los valores de varianza corresponde un componente, que indica la cantidad de unidades que constituyen un porcentaje de varianza. Por ejemplo, el componente 1 puede tener un 20 % de varianza, el segundo componente un 25 %. La cantidad de componentes depende de la matriz original y del tamaño al que se desea reducir la matriz. A cada uno de los valores obtenidos se le conoce como varianza individual e indica la cantidad de información por cada unidad de componente. La cantidad de componentes y varianza adecuados dependen del estudio que se esté realizando, para este trabajo se evalúa la cantidad de componentes acumulados necesarios.
6. Evaluar varianza acumulada. Se utiliza la varianza acumulada que consiste en la suma del total de varianzas individuales obtenidas, por ejemplo, si se utilizaran 2 componentes con varianza acumulada, de acuerdo con el ejemplo anterior, se suma 20 % de la primera varianza y 25 % de la segunda, obteniendo como resultado el 45 % de varianza acumulada, con el fin de determinar la cantidad de información necesaria. (Mazloom y Kasaei, 2005).

Otra posible solución para eliminar datos redundantes es la aplicación de una transformada de *Wavelet* con la aplicación de filtros para dejar un nivel estándar entre los píxeles que se van a procesar y no sea tan variado el valor de los píxeles de acuerdo con la iluminación, pero, como no se asegura la eliminación de píxeles redundantes y además el ruido que contiene la imagen puede incrementar (Ferreira Mejía, Pérez Cubillos, y Aparicio Pico, 2021), es conveniente en este trabajo el uso de PCA.

Es importante identificar los bordes de un objeto en una imagen, que son considerados como el cambio drástico de coloración de píxeles. Uno de los algoritmos empleados para realizar este proceso es el de Sobel, que se enfoca en detectar los bordes en diagonal, horizontal y vertical de una imagen en escala de grises. El algoritmo *Canny* detecta la mayor cantidad de bordes ignorando el ruido que pudiera estar presente en la imagen; el proceso se realiza en tres etapas: filtrado, decisión inicial e histéresis. El ruido en las imágenes se presenta a través de líneas no deseadas, causadas por la cantidad de iluminación o por el color de fondo utilizado para identificar el objeto, (Martínez Susunaga, 2017). Es conveniente eliminar la mayor cantidad de ruido que se presente en la imagen a través de operaciones de acuerdo con la estructura (morfología) del objeto, tales como erosión y el de dilatación, especificado en *Tabla 1*.

A través del filtro de dilatación se agregan píxeles a la imagen de acuerdo con los bordes que presente y de lo que se desee realizar, mientras que el de erosión, elimina píxeles que pudieran ser no deseados, o de diferente tonalidad que sea considerada como ruido (MATLAB, 2021)

*Tabla 1. Operaciones morfológicas*

<b>Filtro</b>	<b>Aplicación</b>	<b>Descripción</b>
<b>Erosión</b>	Elimina píxeles a los límites de los objetos.	Se aplica a imágenes binarias. Elimina los píxeles de menor tamaño con tonalidad oscura. En Python se utiliza la sentencia <code>erode()</code> , que se emplea para realizar la erosión, definiendo el contorno de la figura y haciendo la distinción del fondo por la tonalidad.
<b>Dilatación</b>	Agrega píxeles a los límites de los objetos.	Se aplica a imágenes binarias. Aumenta la región de píxeles blancos. Expande el tamaño de los píxeles con tonalidad blanca y a los píxeles que estén fuera de la región los hace más pequeños. En Python se utiliza la sentencia <code>dilate()</code> , para hacer gruesos los bordes del objeto con tonalidades blancas.



Para desarrollar la Red Neuronal Artificial con Análisis de Componentes Principales, como primer paso es necesario cambiar la dimensión de los datos de entrada para la red neuronal convolucional y conservar la cantidad de píxeles en su mayoría evitando la redundancia a través del uso de la varianza acumulada para que los datos tengan una menor cantidad de dimensiones con la información necesaria disponible, así mismo, con los pasos mencionados se logra conseguir un aprendizaje con un modelo de tres etapas, entrada de datos, entrenamiento y validación, el cual representa un modelo clásico (Rocha Muñoz y Ortiz de Lazcano Lobato, 2016).

Como se ha presentado en la literatura, las variables como iluminación, rotación, tamaño de la pieza y color, son importantes, ya que cambian de acuerdo con el entorno en el que se esté empleando el sistema de reconocimiento, por lo que se utiliza en un entorno diferente, se tiene que adecuar la iluminación para realizar la clasificación con menor cantidad de falsos positivos y negativos.

## **2 PLANTEAMIENTO DEL PROBLEMA**

En algunas situaciones dentro del campo de la manufactura, se requiere de constantes procesos manuales para que el maquinado sea de calidad; sin embargo, pueden producirse errores humanos debido principalmente al “cansancio” del operador, por lo que se recomienda que sea un proceso totalmente automatizado (Kalpakjian y Steven, 2014). Generalmente en los procesos de maquinado, el operador es quien coloca la herramienta de trabajo sobre la pieza metálica y determina el punto de inicio del proceso del maquinado de manera manual, verificando que se desarrolle de manera correcta; estas intervenciones pueden contener error humano (Sánchez J. A., 2006), haciendo difícil la reproducibilidad de cada maquinado, además se necesita mayor tiempo para terminar el proceso. Por otro lado, si se realiza el proceso automatizado, una de las condiciones que afecta es el de la iluminación, porque siempre debe estar presente, pero si las condiciones cambian el sistema causa falsos negativos y positivos, debido a que se deben identificar los bordes de la pieza en tiempo real, entonces se requiere que pueda ser adaptable a la variación de condiciones del entorno de trabajo.

### **2.1 Definición del problema**

El proceso que se lleva a cabo en la manufactura generalmente requiere de la precisión del operador en la intervención para colocar el posicionamiento de la herramienta de trabajo, sobre la pieza de materia prima.

Uno de los aspectos para tomar en cuenta y sustituir la intervención del operador dentro del proceso, es el de determinar en primera instancia la forma geométrica (regular o irregular) de la pieza de trabajo y posteriormente inicie el maquinado. En la literatura no se han encontrado antecedentes relacionados con el presente trabajo, por lo que se opta por comparar la red neuronal convolucional con y sin PCA. Además, una de las principales problemáticas en la automatización del proceso de clasificación de las piezas son las condiciones de iluminación, ya que se debe contar con la intensidad adecuada para lograr que el sistema clasifique las imágenes correctamente, sin embargo, las piezas metálicas por su naturaleza se caracterizan

por poseer tonalidades que ocasionan el reflejo de la luz sobre su superficie, por lo que se dificulta su identificación. Las tonalidades de las piezas metálicas varían en tonalidades oscuras a plateadas, lo que implica otra dificultad ya que la iluminación se debe ajustar para cada una de las piezas (Suárez Castrillón, Alegre, Morala Argüello, y González Castro, 2008).

Una solución es la aplicación de métodos de Visión Artificial, para realizar la clasificación de metales con diferentes colores y materiales ya sean de acero, aluminio, cobre, níquel y hierro, entre otros; al utilizar esta herramienta se requiere involucrar variables como las condiciones de iluminación y cantidad de luxes. Lo anterior implica que, al realizar un proceso de detección de objetos con la ayuda de la visión artificial, se calibrará bajo ciertas condiciones en las que se encuentra el sistema (el margen de variación es estrecho), pero al cambiar de ambiente, las condiciones no serán las mismas y el proceso se verá afectado, requiriendo una nueva calibración o reajuste (Sánchez G. , 2018).

## **2.2 Objetivos de investigación**

### **Objetivo general**

Desarrollar un algoritmo basado en Análisis de Componentes Principales y Redes Neuronales Convolucionales para la identificación de piezas metálicas, que permita obtener una exactitud de al menos 93 %, considerando formas geométricas regulares e irregulares en las piezas, con luz cálida y fría en tres niveles de intensidad lumínica: 1000, 1500 y 2000 lx.

### **Objetivos específicos.**

- I. Desarrollar los algoritmos para la Red Neuronal Convolutiva para clasificar imágenes de piezas metálicas y de Análisis de Componentes Principales para el pretratamiento de imágenes en el entorno de Python.
- II. Aplicar el pretratamiento de imágenes mediante el Análisis de Componentes Principales para reducir la dimensión de la matriz de los datos de entrada en la Red Neuronal Convolutiva.

- III. Verificar el reconocimiento de piezas metálicas mediante tres niveles de variaciones de iluminación establecidos por la Norma Oficial Mexicana NOM-025-STPS-2008, los cuales son: nivel bajo 1000 lx, nivel medio 1500 lx y nivel alto 2000 lx con una fuente de luz cálida y una fuente de luz fría.
- IV. Evaluar el número de neuronas y el número de capas de la red neuronal convolucional necesarias para la clasificación de piezas mediante el diseño factorial  $3^4$  y obtener una exactitud de al menos el 93%.
- V. Construir una interfaz para clasificar piezas metálicas regulares e irregulares en tiempo real o en imágenes contenidas en carpeta para obtener información estadística de la clasificación.

### **2.3 Preguntas de investigación**

¿Es necesario el pretratamiento de imágenes con PCA para que la red neuronal convolucional alcance al menos el 93 % de exactitud?

¿Cuántas neuronas en capa oculta de la red neuronal y qué optimizador, son necesarios para comprobar la exactitud propuesta?

¿Cuál es el porcentaje de varianza y número de componentes en el Análisis de Componentes Principales que permite reducir la cantidad de ruido en imágenes y obtener el mejor porcentaje de exactitud en la clasificación?

¿Con cuál de los niveles de iluminación de 1000, 1500 o 2000 lx se obtiene el mejor porcentaje de clasificación para la métrica de exactitud?

### **2.4 Justificación**

Al desarrollar un algoritmo para clasificar piezas metálicas a partir de la captura de imágenes utilizando cámaras de 1.3 MP y de 8 MP de acuerdo con el intervalo de condiciones de iluminación, se proporciona una herramienta útil para mejorar los procesos de clasificación de piezas metálicas, con al menos 93 % de exactitud. La herramienta permitirá tener una

mayor certeza al determinar se trata de una pieza con forma de rectángulo, cuadrado, círculo, rondana, clavo y tuerca. Las posibles aplicaciones de dicha herramienta pueden ser para la automatización de procesos de maquinado y podrá reducir el error causado por el cansancio del operador.

## 2.5 Impactos

La aportación del desarrollo del proyecto se ve reflejado en los siguientes ámbitos:

- **Científico.** Debido a que en el estado del arte no se ha encontrado información relacionada al trabajo propuesto, se desarrolla el algoritmo de Análisis de Componentes Principales para el pretratamiento de imágenes con variación de iluminación y clasificarlas con una Red Neuronal Convolutiva.
- **Tecnológico.** Se proporciona un algoritmo que puede ser aplicado en la manufactura para la clasificación de piezas metálicas de acuerdo con la forma geométrica regular e irregular, para mejorar la clasificación de al menos 93 % de exactitud y realizar la ubicación de los objetos de manera automática evitando errores humanos.

### **3 HIPÓTESIS**

Con la aplicación del Análisis de Componentes Principales para el pretratamiento de imágenes y el uso de una Red Neuronal Convolutiva, será posible realizar la clasificación de piezas metálicas con formas geométricas regulares e irregulares, con al menos 93 % de exactitud, bajo condiciones de iluminación con luz cálida o fría en tres niveles de intensidad: 1000, 1500 y 2000 lx.

## 4 ESTADO DEL ARTE

Los sistemas de clasificación hacen uso de recursos basados en Inteligencia Artificial como las redes neuronales. Para estos sistemas se establecen variables como pueden ser las formas geométricas para clasificar y la iluminación a la que se somete el objeto al momento de obtener la imagen. La iluminación es una de las variables más importantes ya que depende de ella que se logre identificar la forma del objeto. Las fuentes de emisión de luz que proporcionan la cantidad de iluminación a la que se somete un objeto a clasificar pueden dividirse en natural o artificial; en el caso de la iluminación artificial se pueden encontrar diferentes tipos de iluminación como cenital, la cual emite luz en diversas cantidades y direcciones y la iluminación difusa, en este caso la luz es filtrada por algún objeto; los cuales son importantes a considerar por la forma en la que se ilumina el espacio de trabajo que se utilice debido a las tonalidades que se pueden producir al momento de tomar las imágenes con una cámara. Sin embargo, en ambos casos interviene la intensidad de iluminación medida en luxes, para la reflexión de la luz en los objetos, en especial en piezas metálicas. Otra característica que afecta en la clasificación de objetos con los sensores de cámara, como por ejemplo CCD (Dispositivo Acoplado por Carga) y el sensor de imagen CMOS (Semiconductor Complementario de Óxido Metálico) o CIS (Sensor de Imagen de Contacto), (El Gamal y Eltoukhy, 2005), es el ruido que se visualiza en las imágenes como de sal y pimienta, el cual se refiere a la aparición aleatoria de píxeles de color negro y blanco en la imagen (Menéndez, Javier, López Portilla, Maykel, e Inglesias Martínez, 2013). Por lo tanto es necesario realizar un pretratamiento de imágenes buscando normalizarlas y minimizar el ruido ocasionado por el tamaño en cada las imágenes obtenidas en los dataset, tanto en el de entrenamiento como de validación.

Es importante analizar el efecto que se tiene al clasificar las imágenes normalizadas mediante redes neuronales convolucionales, además, si se emplea otro algoritmo, se pueden comparar los resultados para verificar si se mejoran las métricas como exactitud: evalúa que la etiqueta de predicción sea igual a la etiqueta real, sensibilidad: verifica que los casos positivos sean realmente correctos, precisión: indica el conteo de verdaderos positivos que existen o la

métrica F1 que evalúa la métrica de sensibilidad y precisión para verificar que la red neuronal artificial detecte correctamente las clases y la predicción pertenezca a alguna clase. (Borja Robalino y Monleon Getino, 2020).

En el presente estado del arte se mencionan algunas características de las imágenes que se pueden analizar para mejorar la visualización de la imagen, además, se analiza la literatura acerca de las redes neuronales convolucionales porque son consideradas como las técnicas más adecuadas para la clasificación de imágenes, también se contrasta la información con técnicas como *Wavelet* para el tratamiento de imágenes o el Análisis de Componentes Principales (PCA), para determinar porqué utilizar PCA como técnica de pretratamiento de imágenes en esta investigación. Por otra parte, como es imprescindible analizar los resultados de la red neuronal convolucional, se proporciona información de las métricas exactitud, precisión, sensibilidad y puntuación F1 (métrica que evalúa a la sensibilidad y precisión en conjunto), además de la matriz de confusión para verificar como se clasificaron las imágenes respecto a las clases existentes.

## **4.1 Imagen**

En el proceso de la Visión Artificial se involucran algunos pasos para la adquisición de imágenes tales como: la toma de fotografías, interpretación de datos y el reconocimiento de algunos patrones. Los recursos tecnológicos de hoy en día permiten captar información mediante cámaras y almacenarlas en una matriz (Lira Chávez, 2018).

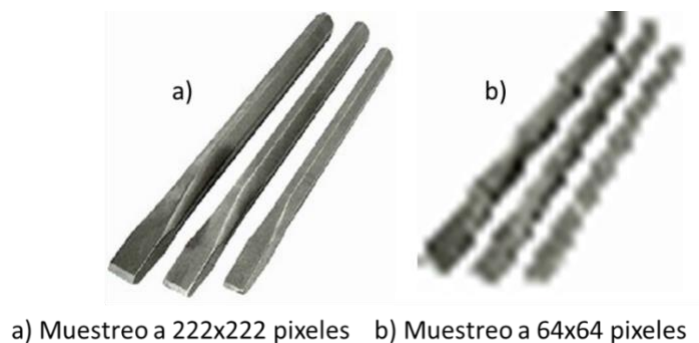
González Marcos (González Marcos, Martínez de Pisón Ascacíbar, y Verónica Pernía, 2006), indica que, para realizar otra forma de representación de una imagen digital, es necesario colocarla mediante una matriz  $f$  de  $N \times M$  dimensiones, indicando la intensidad de color en la imagen por cada píxel, (Figura 3):



$f(1,1)$	$f(1,2)$	...	$f(1,M)$
$f(2,1)$	$f(2,2)$	...	$f(2,M)$
$\vdots$	$\vdots$	...	$\vdots$
$f(N,1)$	$f(N,2)$	...	$f(N,M)$

*Figura 3. Matriz de NxM dimensiones*

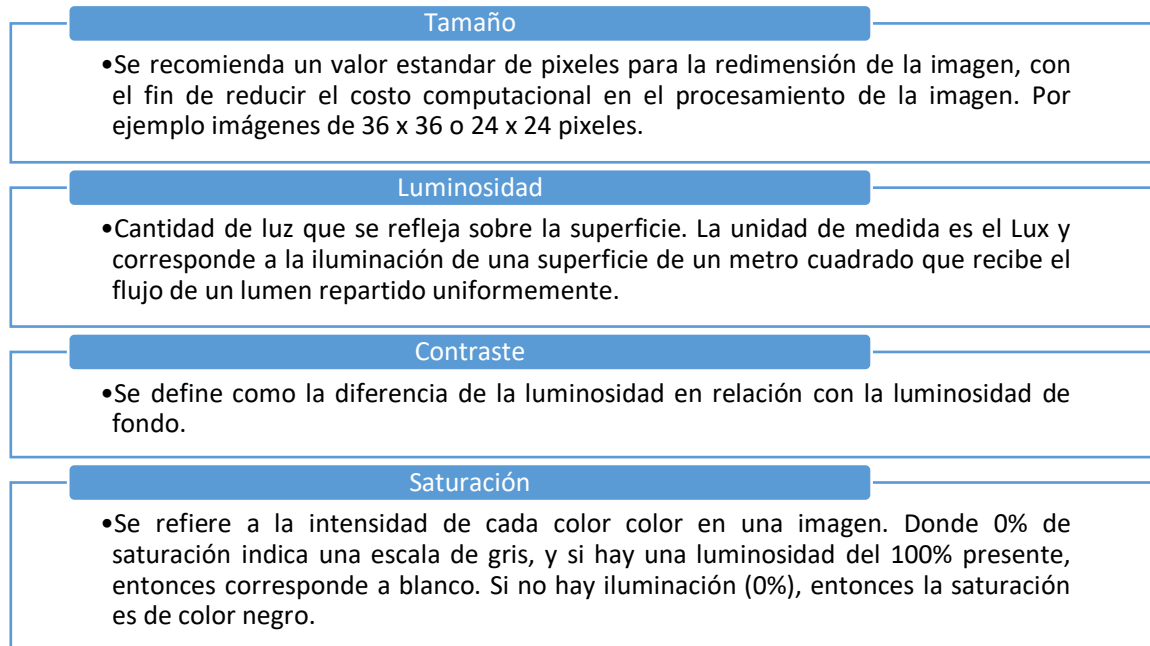
La cantidad de píxeles indica la resolución de la imagen que es capaz de brindar un sistema digital, por ejemplo, cuando se reduce el tamaño de una imagen, se realiza un proceso de muestreo de la imagen, con el motivo de reproducirla en distintas escalas de resolución, pero con un mismo tamaño. Con una imagen original (Figura 4, inciso a) por cada muestra donde hay pérdida de información, así como el ruido, se van generando patrones de forma rectangular (Figura 4, inciso b), (Miranda, 2009; Rosas Echevarria, Solis Bonifacio, y Cueva Cerna, 2019).



*Figura 4. Ejemplo de muestreo*

Aguilar Torres y Bekios Calfa (Aguilar Torres y Bekios Calfa, 2016), mencionan que las imágenes presentan las características mostradas en la Figura 5, las cuales son consideradas

importantes dentro de la Visión Artificial para lograr distinguir un determinado objeto de otro.



*Figura 5. Características de la imagen.*

La imagen puede ser representada por  $n$  bits y depende de un vector formado por tres canales, rojo, verde y azul, también conocido como *RGB*; de la combinación de los colores surge una gama que va de 0 hasta 255 para cada color primario. Por otra parte, cada píxel representa la unidad mínima de la imagen, el cual contiene información del color, brillo y saturación. Si la imagen estuviera filtrada en grises, se consideran tres factores para dejar la imagen en tonalidades de blanco y negro: Alfa para la separación del color rojo y negro, Beta para la separación del color verde y negro; y Gama para la separación del azul y negro. La ecuación correspondiente esta numerada con 1.

$$r = f(x, y) \quad (1)$$

Donde  $r$  es la intensidad de iluminación del píxel en el plano cartesiano  $x, y$ .

Por otra parte, para definir si una imagen está iluminada en las regiones de interés (porque de esto depende un reconocimiento de patrones adecuado), se hace uso de la evaluación del brillo (Flores Eraña, 2009).

La iluminación en una imagen puede provocar sombras, según donde se posicione el objeto de acuerdo con la fuente de iluminación ya sea natural o/y artificial. La cantidad de luminosidad está determinada por la iluminación del espacio. También, para aquellos objetos que reflejan el 80 % o más de la luz aparecerán con alta iluminación y los que reflejen al menos 3 % tendrán apariencia opaca (Aguilar Domínguez, Mejía Lavalle, y Sossa, 2021).

La imagen de la Figura 6, presenta algunas áreas con mayor iluminación, otras sombreadas como la parte que se señala con el cuadro de color naranja. Esto causará que los bordes del objeto no sean reconocidos de acuerdo con el área original y se descarten regiones importantes de la imagen que pueden ser utilizadas como características, entonces, el sistema podría interpretar que el fondo es parte del objeto, lo cual es incorrecto.



*Figura 6. Imagen con brillo; el recuadro en naranja ocasiona problemas en la identificación.*

Se puede evaluar también el tono del objeto, se considera tono a las características para distinguir el color, por la cual designamos los colores: verde, violeta, anaranjado, entre otros colores que pueden resultar de la combinación del *RGB*. El tono sirve como indicador de si el píxel parece similar a un color o a una porción. El color se asocia a la percepción de una cantidad de luz determinada porque los objetos reflejan un porcentaje de luz no absorbida que es lo que generalmente percibe el ojo humano y corresponde a la información que se brinda a los sistemas de visión artificial (Yang, 1993).

Filipo Rugeles (Filipo Rugeles, Cano Garzón, y Chavez Osorio, 2010), señala que la luminosidad también depende del tipo de fuente de dónde provenga, como se indica en la Figura 7. Hay tres tipos de iluminación: natural, artificial y global, que se pueden detectar al momento de realizar el reconocimiento de patrones, dependiendo de las pruebas que puedan realizarse, aunque, resulta de mayor importancia el estudio de la iluminación de manera global, porque intervienen las variaciones de iluminación que son las que afectan en la discriminación de los Eigenvalores (o descriptores) de los objetos a reconocer. También porque usualmente se trabaja con iluminación tipo LED debido a que son considerados como lámparas ahorradoras de luz y se tienen presentes al momento de hacer uso de algún sistema de visión artificial. La iluminación con LED proporciona iluminación fría, lo cual afecta en la tonalidad de la pieza al momento de clasificarla, en ocasiones provoca que se torne muy brillante o que los bordes de las piezas no se reconozcan de manera correcta, por lo tanto, es importante considerar que tipo de iluminación utilizar, ya sea iluminación fría o cálida, según el tono de las piezas u objetos.

Natural	<ul style="list-style-type: none"> <li>•Depende de los factores climáticos</li> <li>•Es variable en cantidad de iluminación y posición de sombras</li> </ul>
Artificial	<ul style="list-style-type: none"> <li>•Son producidas por lámparas incandescentes: halógedas, LED.</li> <li>•Presenta variación con la tensión nominal, generando variaciones en el flujo luminoso</li> </ul>
Global	<ul style="list-style-type: none"> <li>•Unión de la luz natural con la artificial</li> </ul>

*Figura 7. Tipos de iluminación.*

De acuerdo con los niveles de iluminación que proporcionan las lámparas y que reflejan los objetos, de los cuales se adquieren imágenes, se recomienda realizar la obtención del píxel más alto mediante el cálculo de percentiles, a través de la ecuación (2), para poder aplicar un balance de iluminación (Rendón Macías, Villasís Keeve, y Miranda Novales, 2016):

$$Percentil = \frac{Posición\ del\ dato * número\ de\ elementos\ de\ la\ muestra}{n} \quad (2)$$

Donde  $n$  es el total de la muestra de píxeles. Después de obtener el percentil, se normalizan los valores mediante el rango de píxeles mayores y se realiza un filtro de los valores menores.

La iluminación y percentiles mencionados anteriormente involucran a las imágenes, sin embargo, al utilizar sistemas de visión artificial para clasificar objetos, intervienen otros factores como guiado, inspección y tipos de iluminación para detectar la ubicación del objeto y realizar la predicción.

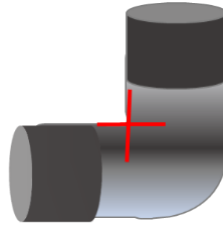
## **4.2 Factores que intervienen en Visión Artificial**

En el ámbito de la clasificación de objetos se menciona el siguiente ejemplo, donde determinadas piezas metálicas que pasan a través de una banda transportadora que se mueve a una velocidad a la que un operador no puede determinar tan fácilmente de que pieza se trata, se requiere que el sistema de clasificación sea rápido y preciso para detectar los patrones de información, entonces, para lograr la identificación de objetos es primordial la localización de la pieza, lo cual se puede lograr a través del guiado, pero no solo interviene ese factor, sino también se evalúa la calidad de las piezas a través de la inspección, por lo tanto se debe contar con un tipo de iluminación adecuada (Londoño Osorio, Marín Pineda, y Arango Zuluaga, 2013). A continuación, se definen los factores: Guiado, Inspección e Iluminación.

### **4.2.1 Guiado**

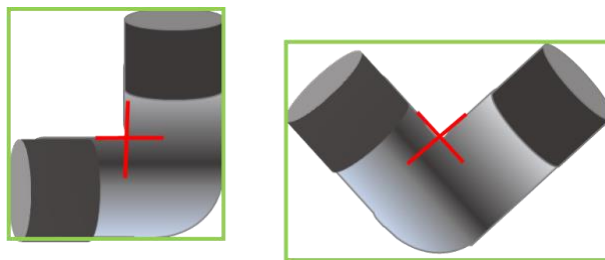
El guiado principalmente se utiliza para localizar la pieza y la orientación de esta en un espacio 2D o 3D, con el fin de que un robot o una unidad de control, pueda manipular la pieza. Generalmente se utiliza una cruz como orientación de donde se va a aplicar la acción correspondiente, o para indicar el inicio del maquinado. Las ventajas del guiado son la obtención de una velocidad y una precisión superiores al posicionamiento manual para el embalaje, búsqueda y alineación de piezas en el montaje de otros componentes, localización de piezas en un estante o retirar piezas de los almacenes (Riveros Guevara, Salas López, y Solaque Guzmán, 2012).

El guiado es un factor importante en Visión Artificial porque la presentación de las piezas hacia la cámara puede ser con orientaciones desconocidas. Otra ventaja de la aplicación del guiado es el posicionamiento automático de las herramientas (Figura 8), implicando características claves como: calibre, nivel, perfil u otros descriptores.



*Figura 8. Codo de 90°*

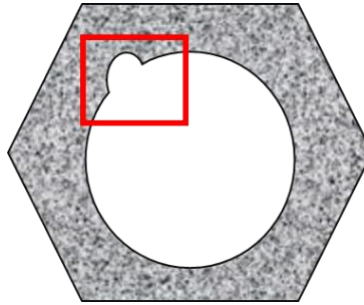
El guiado puede depender de algunos patrones geométricos, por lo que se debe considerar la tolerancia en las variaciones en el contraste e iluminación, así como cambios de escala y de rotación para lograr el reconocimiento de la misma pieza a pesar de estar en una localización distinta (Figura 9) (Riveros Guevara, Salas López, y Solaque Guzmán, 2012).



*Figura 9. Identificación de una pieza en distinta posición.*

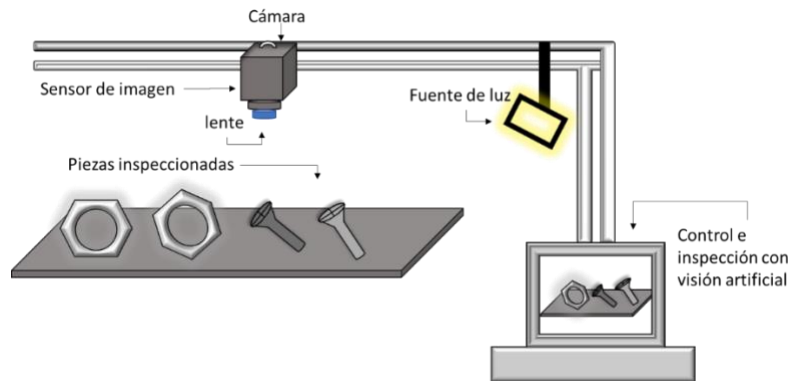
#### **4.2.2 Inspección**

Usualmente se utiliza el sistema de visión artificial para detectar defectos (Figura 10), imperfecciones funcionales e irregularidad propiciados por la manufactura, para garantizar la integridad de los productos ya sea de paquetes de industrias, farmacéutica, anillos de botellas que son plásticos o metales que van alrededor de la boca de la botella, entre otros (Rojas Hernández, Silva Ortigoza, y Molina Vilchis, 2007).



*Figura 10. Identificación de irregularidades.*

Un sistema de visión artificial para la identificación de patrones puede estar conformado por algunos elementos (Figura 11) como la fuente de iluminación, una lente, sensor de imágenes, algoritmo para el procesamiento de las imágenes para la extracción precisa de información, inspección necesaria y como resultado obtener una decisión proporcionada por el sistema de clasificación (Cognex, 2020).



*Figura 11. Componentes de un sistema de Visión Artificial, adaptado de (Cognex, 2020).*

De acuerdo con Pérez Lara, Ochoa Somuano, Pereyra Ramos, y Toscano (Pérez Lara, Ochoa Somuano, Pereyra Ramos, y Toscano Martinez, 2013), una de las variables que interviene en el reconocimiento de objetos es la iluminación, sin embargo, existen diferentes tipos, como iluminación de campo oscuro, cenital difusa, estroboscópica, las cuales se utilizan de acuerdo a las condiciones que se requieran en el estudio que se vaya a realizar.

### **4.2.3 Iluminación**

Los sistemas de visión artificial generan proyecciones a través de los datos obtenidos mediante la cámara y la cantidad de luz que se refleja sobre el objeto. Las técnicas de iluminación que se emplean consisten en una fuente de luz y la ubicación en relación con el objeto, con el fin de conservar la forma del original sin anular los rasgos que presentan al momento de capturar la imagen con la cámara.

#### **Iluminación de campo oscuro**

Se utiliza para evaluar los bordes, surcos o grietas de objetos que contengan algún defecto superficial y con ello mejorar el contraste en donde la visualización a simple vista sea difícil.

#### **Iluminación cenital difusa**

Este tipo de iluminación es propagada en cualquier dirección y no genera sombras tan notorias. Un ejemplo es el de la bombilla ya que produce iluminación que es difundida en distintos ángulos dentro de un espacio, además, proporciona una iluminación cálida, haciendo que se suavicen las tonalidades de los objetos presentes.

#### **Iluminación estroboscópica**

La iluminación estroboscópica se usa en aplicaciones donde se aplica una velocidad para inmovilizar objetos que se están moviendo durante su examen. El uso de una luz estroboscópica también ayuda a evitar imágenes borrosas.

Para considerar los parámetros anteriores de un sistema de visión artificial, se toma en cuenta que primero se tiene que adquirir la imagen mediante una cámara, la cual depende de una determinada calibración (posicionamiento de acuerdo a los ejes  $x$ ,  $y$ ,  $z$ ), para después, obtener una imagen, de acuerdo con el ángulo de posicionamiento de la cámara con respecto a la imagen, y finalmente realizar el procesamiento de imágenes a través de clasificadores convencionales y clasificadores profundos (Calvete Sanz, 2018).



### 4.3 Clasificador: Red Neuronal Artificial (RNA)

Salazar Cárdenas y Vargas Trujillo (Salazar Cárdenas y Vargas Trujillo, 2017), señalan que las Redes Neuronales Artificiales son una representación del funcionamiento de una neurona de acuerdo con su estructura biológica: cuerpo celular con una rama principal llamada Axón que se encarga de recibir las señales eléctricas para distribuirla entre sus terminales, es decir, otras ramas más cortas denominadas dendritas.

Las neuronas están interconectadas entre sí, para lograr procesar información, a través de un entrenamiento ya sea supervisado: para obtener una salida deseada a partir de datos previamente tratados; o no supervisado: ocurre cuando se pretende que la red neuronal “comprenda” por sí misma un conjunto de datos, al igual que su estructura.

El modelo de una Neurona Artificial se representa en la Figura 12, la cual funciona a partir de la obtención de las entradas: número de entradas que recibe, correspondiendo a la información a procesar, pesos sinápticos: atenúan o amplifican los valores de entrada para distribuirlos en la neurona, una función de activación y una salida (Fernandez, Guzman, y Dadios, 2018).

En el proceso de la suma de los pesos, se recibe el parámetro  $\theta_i$ , donde  $\theta$  son los pesos e  $i$  pertenece al *bias*, también llamado umbral de la neurona, luego el resultado de la suma se le denomina entrada neta, con la cual se determina si la neurona esta activa o no (función de activación). La activación de neuronas surge cuando se suman las señales recibidas, entonces la salida se envía a la neurona y si está por debajo del umbral, la neurona no se dispara, en otras palabras, no genera ninguna señal; la función del umbral que decide si se dispara la neurona se le conoce como función de activación, para que después, la información sea propagada entre las demás neuronas (Garro, Sossa, y Vázquez, 2012).

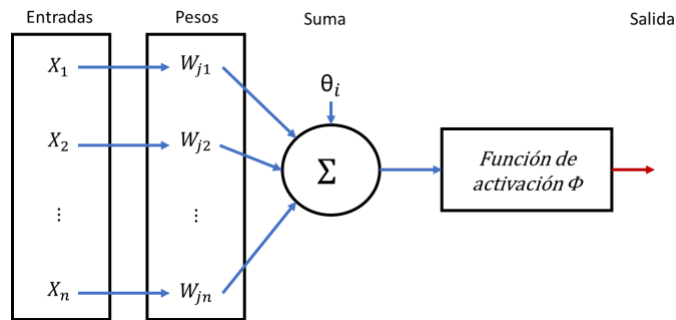


Figura 12. Modelo de una Neurona Artificial, adaptado de (Salazar Cárdenas y Vargas Trujillo, 2017)

Delgado Rivero (Delgado Rivero, 2001), aborda que una Red neuronal está constituida por una arquitectura en capas, una de ellas es la capa de la cual se pueden realizar predicciones, otra capa es la intermedia (la cantidad depende del tipo de red neuronal artificial que se esté utilizando) en la que se realiza el procesamiento y se les denomina capas ocultas, por último, se tiene la capa de salida.

Los nodos son unidades individuales que procesan la información de cada capa implementando funciones de activación. Cada uno de estos nodos se encuentran interconectados para compartir la información entre ellos. La importancia de esto es que en cada interconexión existe un peso, que especifica la fuerza de interconexión entre dos nodos, para especificar el modelo de entrenamiento de los datos ingresados; el objetivo es el de lograr realizar una tarea como detección de objetos, clasificadores, entre otros, (Zambrano Escobar, 2009).

Una Red Neuronal Artificial *backpropagation*, está destinada para resolver problemas de reconocimiento de patrones y un perceptrón aprende teniendo en cuenta solo el error entre el objetivo, el resultado y la tasa de aprendizaje (Cervantes Osornio, Arteaga Ramírez, Vázquez Peña, y Ojeda Bustamante, 2016). Aziz (Aziz, Salam, Bin Haji Salam, Ullah Sheikh, y Ayub, 2020), menciona que estos modelos, aplicados en la detección y clasificación de objetos, utilizan métodos que les ayudan al aprendizaje automático para determinar lo siguiente:

- Seleccionar la región informativa: proceso altamente computacional que permite determinar la identificación del objeto.
- Extracción de características: se basa principalmente en la observación natural de los objetos para determinar la naturaleza y robustez del objeto. Algunas técnicas para extraer características son: *Scale-Invariant Feature Transform (SIFT)*, Histograma de Gradientes Orientados (*HOG*), *HAAR*: Clasificador para la detección de objetos.
- Clasificación: categorizar al objeto de entrada, según las posibles categorías de destino.

De esa manera se logra el objetivo de identificación de patrones, sin embargo, muchas veces en las regiones informativas de las imágenes, intervienen elementos que dificultan la extracción de la región de interés, para lo cual, se utilizan otros métodos (*HOG*, *HAAR*, *PCA*) para lograr la discriminación y obtención correcta del elemento que se requiere, contenido generalmente en una imagen o fotografía.

A continuación, se explica de manera más detallada el funcionamiento de una Red Neuronal Convolutiva para la detección de objetos, así como también, el desarrollo de las capas que la determinan.

#### **4.3.1 Red Neuronal Convolutiva**

El principal objetivo de una Red Neuronal Convolutiva es extraer y clasificar diferentes objetos de acuerdo con las características que se obtienen a partir de una base de datos de imágenes. Las aplicaciones de la RNC se ven reflejadas, por ejemplo, en la detección de dígitos escritos a mano y en la detección de rostros, teniendo como datos de entrada una base de datos de imágenes extensa (Rivas Asanza y Bertha, 2018).

La extracción de características locales se realiza a alta resolución, lo que minimiza la pérdida de información, además se tiene como alternativa, la combinación con resoluciones bajas (72 DPI, en español Puntos Por Pulgada), con el fin de utilizar funciones más complejas, en caso de que se deseen implementar (Camacho y Labrandero, 2016).

La estructura de este tipo de arquitectura consta de una entrada de longitud variable, que se refiere a la entrada de datos, por ejemplo, una cadena u oración que tiene 12 palabras debe ser representada mediante una variable por cada una de las instancias creadas, por consecuente de tener una entrada de datos, a los cuales, se les debe aplicar una operación interna relacionada con la memoria del cálculo, denominadas como capas ocultas (Cruz Roa, Suárez, Jiménez L, y Castro Franco, 2017).

Para la salida de datos, en relación con la estructura, se utiliza una salida de longitud variable, esto quiere decir que, por ejemplo, si se quiere traducir un texto del español al inglés, la entrada de datos es variable, luego, la longitud de secuencia de salida es diferente a la de entrada (Delgado Rivero, 2001).

A razón de lo anterior, se utilizan capas para poder organizar la información: capas convolucionales (capas C), asociadas con celdas simples y capas de agrupación (capas P) para modelar el comportamiento de células complejas. La capa C realiza una convolución en 2D. Mientras que las capas de agrupación reducen el tamaño de la entrada mediante un barrido espacial (Oludare Isaac, Jantan, y Dada, 2019).

Según Scherer (Scherer, Müller, y Behnke, 2010), en la Figura 13, se presenta la arquitectura de la RNC. Una de las características que poseen y con las cuales se puede diferenciar de las RNA clásicas, son las operaciones de convolución para realizar el procesamiento de la imagen y desglosarla en imágenes más pequeñas para reducir el tamaño inicial y que sean filtradas a través de procesos como *max* o *min pooling*, para que como resultado final se obtenga la clasificación de la imagen de entrada, respecto a las que clases con la que se haya entrenado la Red Neuronal Convolucional.

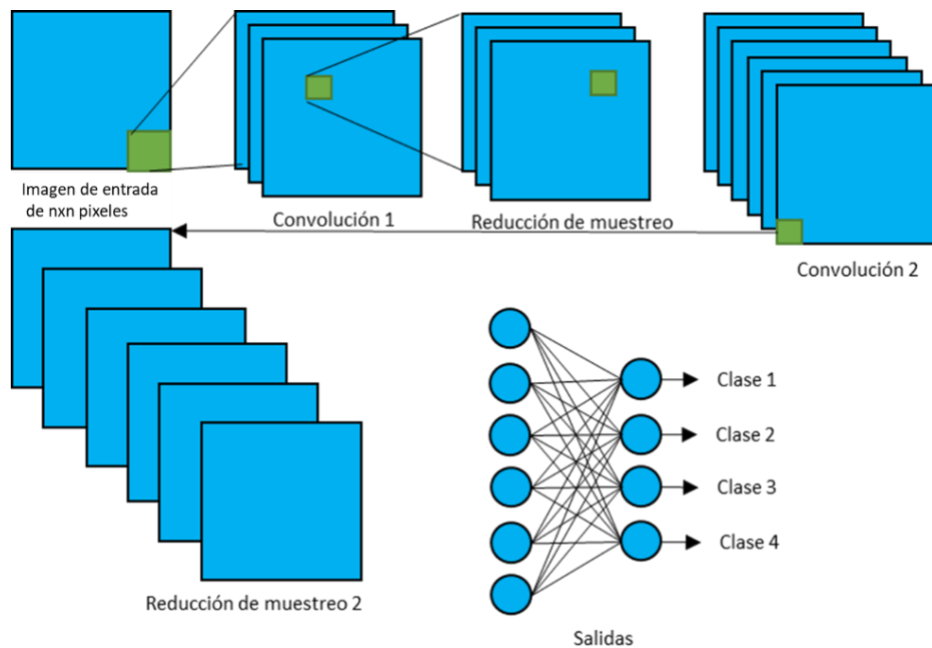


Figura 13. Etapas de la Red Neuronal Convolutiva, adaptado de (Scherer, Müller, y Behnke, 2010)

El filtro no se aplica en toda la imagen ( Figura 14), sino que se realiza por partes para que el procesamiento de la imagen en la red neuronal tenga menor costo computacional y sea rápida la filtración respecto a filtrar toda la imagen. Las primeras capas están destinadas a identificar patrones básicos, por ejemplo: líneas rectas o curvas. Luego por cada salida se analizan otros patrones cada vez más complejos, hasta llegar a la identificación de un objeto (Khan, Rahmani, Shah, y Bennamoun, 2018).

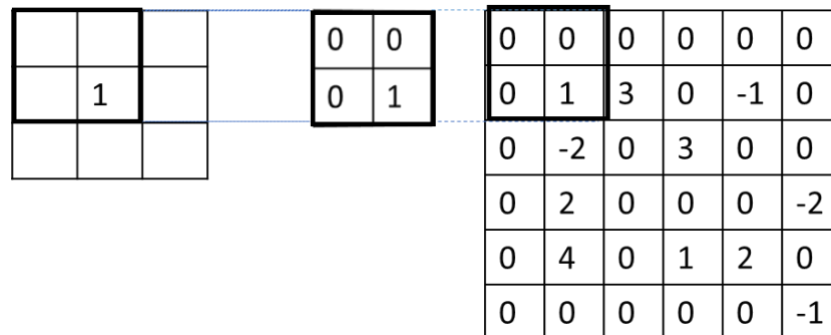


Figura 14. Aplicación de filtro, (Khan, Rahmani, Shah, y Bennamoun, 2018)

Para mejorar el funcionamiento de la Red Neuronal Convolucional, se utilizan métodos de extracción de características, los cuales se dividen en dos grupos: superficiales y profundos.

Los superficiales ignoran los detalles de las formas que describe de manera efectiva a la imagen, mientras que los profundos, se encargan de extraer características de oclusión del objeto, así como también, los puntos clave, a través de descriptores: *HOG*, *SURF*, *FREAK*, que se mencionan por ser más utilizados en aplicaciones de visión artificial por computadoras con Redes Neuronales Convolucionales (Cruz Roa, Suárez, Jiménez L, y Castro Franco, 2017).

Resendiz Tolentino (Resendiz Tolentino, 2017), señala que normalmente las operaciones de convolución son utilizadas para el tratamiento de imágenes y para el tratamiento de señales, sugiriendo que para el filtrado de imágenes, se puede utilizar como método la transformada de *Wavelet* o de Fourier, sin embargo, la transformada de *Wavelet*, aunque se enfoque en el filtrado de las imágenes, carece de precisión y tiene restricciones de acuerdo con los filtros (pasa bajas, pasa altas) que se manejen, los cuales limitan la frecuencia y la resolución temporal en la imagen. Además, se deben considerar parámetros como frecuencias altas o bajas, ejes de tiempo, lo que impide que se demuestre una claridad en la señal, generando una restricción de la incertidumbre en la resolución.

Para el filtrado o matriz de convolución, de una imagen, se tiene que tomar en cuenta la descomposición en  $X$  y  $Y$  pixeles, correspondiendo al alto y ancho, puesto que son variables importantes para el caso de tener y mantener una buena resolución en el tratamiento de imágenes en dos dimensiones, ya que ayuda a tener un buen pre tratamiento de datos, debido a que los pixeles de las imágenes forman una matriz con datos no estandarizados, derivados del brillo, iluminación, factores ambientales en el momento de la toma de la fotografía, sin embargo, no siempre se hace una selección adecuada del área de interés, para lo cual, se utilizan métodos como los ya mencionados (*HAAR*, *SIFT*, *HOG*), además, se puede emplear, análisis de componentes principales (Ali Ahmed, Dey, y Kumar Sarma, 2011).

Para el análisis de bases de datos de imágenes extensas es conveniente utilizar una Red Neuronal Convolutiva, a través de un tratamiento previo a los datos de entrada para la obtención de matrices, para clasificar y localizar imágenes con menos cantidad de parámetros de entrada, porque reduce la dimensionalidad de las características de las imágenes, a través de operaciones convolucionales (Islam, Rahman, Siddique Nafiz, y Jabid, 2018).

Para comparar la arquitectura de las redes neuronales, en relación con las características, ventajas y desventajas que presentan se hace uso de la Tabla 2-3. El motivo es demostrar la viabilidad de las Redes Neuronales Convolucionales (RNC) en la clasificación de objetos a partir de imágenes de entrada para el entrenamiento de acuerdo con Alsmadi (Alsmadi, Omar, Noah, y Almarashdah, 2009).

Tabla 2. Comparación de los tipos de arquitectura de redes neuronales convencionales.

Nombre de la red neuronal	Características principales	Ventajas	Desventajas
<p><b>Perceptrón</b> Tipo de aprendizaje: <b>supervisado.</b></p> <p>Se usa para realizar separaciones estadísticas. Clasificación de patrones correctos.</p>	<p><b>Simple:</b> Es considerada como el tipo simple de una Red Neuronal Artificial, porque tiene una alimentación directa en las neuronas de entrada. Presenta una tasa de convergencia lenta y búsqueda de mínimo global</p>	<p>Umbral y pesos sinápticos ajustables. Puede usarse como clasificador a partir de un <i>dataset</i> con etiquetas.</p>	<p>Utiliza una sola neurona de salida. El entrenamiento del perceptrón debe extraerse de clases linealmente separables. Se limita a realizar la clasificación de patrones con solo dos clases. No todas las funciones lógicas pueden realizarse, ejemplo: XOR, no linealmente separable.</p>
	<p><b>Multicapa:</b> Mapea los datos de entrada en salida apropiada. Utiliza 3 o más capas de neuronas (nodos), con funciones de activación no lineal. Distingue datos no linealmente separados. Conjunto de unidades de procesamiento como células. Cada una de las conexiones de neurona a neurona es determinada por un peso, el cual genera una respuesta sobre ella.</p>	<p>Es capaz de crear modelos no lineales. Se puede entrenar un modelo en tiempo real. La propagación define la entrada segura de una neurona a partir de datos externos, logrando distinguir datos separados.</p>	<p>Modelos con diferente precisión por presentar pérdida en el mínimo local, debido a la asignación aleatoria de valores. Son sensibles a la escala de datos, es decir, que tengan un valor medio e idéntica desviación estándar.</p>
<p><b>Adaline</b> Tipo de aprendizaje: <b>supervisado.</b></p>	<p>Su función se determina a partir de la sumatoria de las entradas que generan 0 y 1, que depende del umbral que se esté manejando El algoritmo de aprendizaje es RMS (Error cuadrático medio). Se usa para resolver problemas de regresión lineal y linealmente separables.</p>	<p>Se pueden usar varias arquitecturas del tipo Adaline para formar una arquitectura más grande, conocida como Madaline.</p>	<p>No existen reglas sistemáticas que puedan describir el problema.</p>



Nombre de la red neuronal	Características principales	Ventajas	Desventajas
<b>Feedforward Networks (Redes de retroalimentación)</b> Aprendizaje <b>supervisado</b> o <b>no supervisado</b> . Se utiliza para estimar patrones y cuando el flujo de la información sea en una sola dirección.	Llamadas perceptrones multicapa de retroalimentación. El objetivo es aproximar alguna función. La primera capa identifica bordes, en la segunda capa oculta se identifican líneas externas del objeto.	Su ejecución es rápida.	No existe una retroalimentación donde las salidas se realimentan entre sí.

*Tabla 3. Comparación de los tipos de arquitectura de redes neuronales de aprendizaje profundo.*

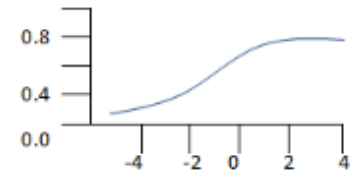
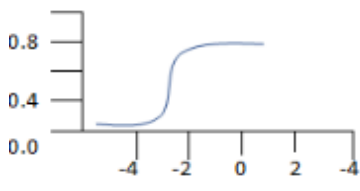
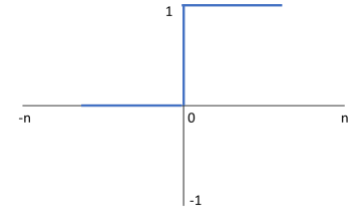
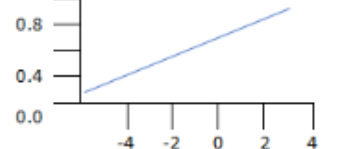
<b>Redes neuronales de aprendizaje profundo</b>			
Nombre de la red neuronal	Características principales	Ventajas	Desventajas
<b>Redes Convolucionales Artificiales (RNC)</b> Tipo de aprendizaje: <b>supervisado</b> . Comúnmente usado para clasificar y detectar objetos, caras, patrones, etc. Identifican patrones gráficos y luego realiza una clasificación a partir de estos.	Se utilizan para el análisis de imágenes, debido a que son capaces de detectar características simples como por ejemplo detección de bordes y líneas para componer de acuerdo con características más complejas hasta detectar lo que se busca. Son las más utilizadas para la detección de imágenes por la ventaja de reconstruir imágenes a través de la segmentación por convolución y la aplicación de filtros hasta determinar la clase respectiva. Aplica una reducción de las matrices obtenidas de las imágenes, generando un mapa de características con kernel (filtros).	Reduce el número de conexiones y parámetros de entrenamiento. Al reducir los parámetros, solo quedan las características más comunes. La capa de salida tiene la misma cantidad de neuronas que de clases.	En la reducción de características se pierde información relevante de las imágenes analizadas, por lo tanto, pierde precisión. La cantidad de datos determina la cantidad de neuronas.

<p><b>Red Neuronal Artificial Backpropagation</b>  Tipo de aprendizaje:  <b>Supervisado.</b>  Es funcional para realizar algoritmos de optimización porque se ajustan los pesos para mejorar la tasa de aprendizaje.</p>	<p>La búsqueda de los mínimos globales puede quedar atrapada en los mínimos locales y el algoritmo puede ser sensible a los parámetros seleccionables por el usuario.</p>	<p>Usa patrones de entrenamiento para realizar la clasificación.  Necesita de una etapa de aprendizaje.  Permite entrenar redes multicapa.  Exige derivar la función de activación.</p>	<p>Requieren un valor mínimo de error.  Los pesos van tomando valores que pueden ser grandes y la red aun así no cambia.  Lentitud del aprendizaje.  Las capas ocultas pueden generar alteraciones en la salida.</p>
<p><b>Máquina de Boltzmann</b>  Tipo de aprendizaje:  <b>supervisado o no supervisado</b>  Tiene una estructura de tipo recurrente para operar de manera binaria.</p>	<p>Su funcionamiento es a través de valores binarios.  Es probabilístico.  Unidades ocultas o no  Conexiones bidireccionales.  Pueden ser secuenciales: cambian de un estado a la vez.  Pueden ser paralelas: cambian de estado de manera simultánea.</p>	<p>Arquitectura simple  Son utilizadas para distribuciones de tipo probabilísticas de vectores arbitrarios.  Calcula la probabilidad de vectores binarios.</p>	<p>Difícil de entrenar y lenta para el aprendizaje  Conectividad restringida.</p>
<p><b>Redes recurrentes</b>  Funcionan como memorias asociadas.</p>	<p>Usa datos temporales  Presenta menos compatibilidad con funciones.  Las secuencias de entradas arbitrarias.  Proporcionan información de series de tiempo.</p>	<p>Son tolerables al ruido.  Es un proceso adaptativo por la realización del cálculo de matrices.</p>	<p>La red se vuelve inestable si existen patrones parecidos  El número de patrones a aprender son limitados por el número de nodos en la red.</p>

De las arquitecturas anteriores, se utiliza una Red Neuronal Convolutiva por la capacidad de procesar grandes cantidades de imágenes, ya que son las más adecuadas para la clasificación de objetos.

Para que las redes neuronales obtengan una salida a partir de los valores de entrada, y que la respuesta se encuentre en un rango por ejemplo de (0,1) o (-1,1), se utilizan funciones de activación. Las ecuaciones que se utilicen en la función de activación se relacionan con una gráfica de salida, que permite el análisis de datos; las más representativas se observan en la Tabla 4 (Soares y Souza, 2016):

*Tabla 4. Principales funciones de activación para el análisis de datos en Redes Neuronales.*

Función	Aplicación	Ecuación	Gráfica
Sigmoidal	Satura el gradiente. Lenta convergencia. No está centrada en el cero. Está acotada entre 0 y 1. Buen rendimiento en la última capa.	$f(x) = \frac{1}{1 + e^{-x}}$	
Tangente hiperbólica	Muy similar a la sigmoide: Satura y mata el gradiente. Lenta convergencia. Centrada en 0. Está acotada entre -1 y 1. Buen desempeño en redes recurrentes.	$f(x) = \frac{1 - e^{-x}}{1 + e^{-x}}$	
Función de transferencia Hardlim	Conocida como escalonada. La neuronal de salida es 1 en caso de ser mayor que el umbral, si no, el valor es 0.	$f(x) = \begin{cases} 1 & \text{si } x \leq 0 \\ 0 & \text{para lo contrario} \end{cases}$	
Lineal	Dónde x es una constante.	$f(x) = x$	

--	--	--	--

#### 4.4 Análisis de Componentes Principales

Los componentes principales de una imagen son los descriptores que contienen información de acuerdo con la dimensionalidad de los datos; los cuales están ubicados en un plano cartesiano y posteriormente se colocan dentro de una matriz para redimensionarla en una nueva matriz de menor tamaño, evitando la pérdida de información importante por medio de las características principales de la imagen. Con lo anterior se tiene la posibilidad de hacer una reconstrucción del objeto a analizar a través de una proyección. Una vez que sean graficados estos resultados, se procede a la siguiente etapa que corresponde a la designación de etiquetas, que consiste en que al patrón obtenido en la graficación se le va a designar un nombre el cual va a usar como identificados para los demás objetos (Gitesh Chitaliya y Trivedi, 2010).

El algoritmo de PCA tiene tres secciones importantes: obtención de la media de los datos, matriz de covarianza y definición de componentes, por lo tanto, se siguen los siguientes pasos:

1. Se obtiene la media de los datos se obtiene y se resta a cada una de las características ( $x_i$ ) mediante la ecuación (3), con el fin de normalizar los datos.

$$\text{normalización} = \frac{x_i - \bar{x}}{\sigma(x)} \quad (3)$$

Donde  $x_i$  es la primera característica extraída que se resta de la media ( $\bar{x}$ ) de la suma de todas las características de la imagen.  $\sigma$  representa la desviación estándar de la variable  $x$ .

2. Se obtiene una matriz que contiene las imágenes de entrada, donde  $X$  es el conjunto de imágenes de entrada (Naveed Abbasi y He, 2019). Para calcular un valor propio ( $\lambda$ ) se utiliza la ecuación (4)

$$B \cdot \vec{a} = \lambda \cdot \vec{a} \quad (4)$$

3. La matriz original ( $B$ ) tiene dimensión cuadrada y un Eigenvector ( $\vec{a}$ ). Con los valores propios obtenidos, se calcula la varianza total de todos los componentes también conocida como varianza acumulada (5) (Naveed Abbasi y He, 2019).

$$\sum_{i=1}^p Var(x_i) = \sum_{i=1}^p \lambda_i \quad (5)$$

$\lambda_i$  son los valores propios. Una vez obtenidos los valores propios, se generan las nuevas coordenadas correspondientes a los componentes. Para reducir las dimensiones de la varianza obtenida, se calcula el número de componentes a partir de la mayor varianza ortogonal. La varianza consiste en la combinación de variables originales, como el número de pixeles, las líneas, las curvas, según las características de la imagen.

4. Para calcular un componente, se resta el valor de la media de las mismas variables que componen la imagen. Así, se consigue una media de cero en todas las componentes.
5. Por último, se calcula la covarianza de los vectores propios. Al conjunto de descriptores se les llama Eigenectores. Se les aplica una normalización, para estandarizar el rango de los pixeles y no presentar perdida de información por dos casos: naturaleza de los pixeles de la imagen y el rango de intensidad luminoso limitado. (Salgado Díaz del Río, 2019).

Del resultado obtenido se evalúan las características de las imágenes como la dimensión y cantidad de pixeles, a través de la implementación de un conjunto de filtros como el de la normalización o *K-Vecinos Cercanos*. Con la aplicación de dicho filtro se mejoran las características del conjunto de datos de selección que constituyen la matriz de pixeles (Reyes Nava, 2018).

Se recomienda pasar la imagen original a escala de grises, posteriormente, utilizar un histograma que indica la distribución de la escala de grises y los niveles que presentan para analizar de los colores de la imagen. (Mathworks, 2021).

Las imágenes pueden presentar opacidad y ocasionar una mala clasificación de objetos, entonces, se requieren evaluar parámetros de la imagen como: iluminación, contraste y brillo, se utiliza un histograma en el cual resaltarán las áreas más brillantes en alto contraste y las sombras se verán más oscuras (Esqueda Elizondo y Palafox Maestre, 2005). Por ejemplo, en la Figura 15 se muestra la imagen original a), a la cual se le aplicó el filtro en escala de grises b). De la imagen en escala de grises se obtuvo un histograma que se divide en tres partes. La primera región empieza de 0 a 75 que hace referencia a la

intensidad de iluminación que representa a las sombras de la imagen. De 76 a 175 pertenece a la luminosidad media y de 176 a 255 indica las zonas de luces.

Los valores más altos en el histograma de la imagen Figura 15,b) del eje de cantidad de píxeles, corresponde a la primera región con una frecuencia de 1.2 píxeles, indicando que la imagen está subexpuesta, es decir, que la imagen está demasiado oscura, que carece de luces y que es una imagen de bajo contraste. La flecha roja indica la región en donde existe un bajo contraste en la imagen y en el pico que está a la derecha de la flecha se encuentran las luces que contiene la imagen.

La comparación de los histogramas es que en la imagen a color Figura 15 a), se encuentran diferentes tonos, representados de manera digital por la composición *RGB* (rojo, verde y azul), los colores que puede tomar la imagen van en una escala de 0 a 255, por lo tanto, es necesario normalizar estos conjuntos de valores en tonos grises en escala de 0 a 255, imagen Figura 15 b), para tener una sola gráfica que indique en escala de grises, la manera en la que está iluminada la imagen.

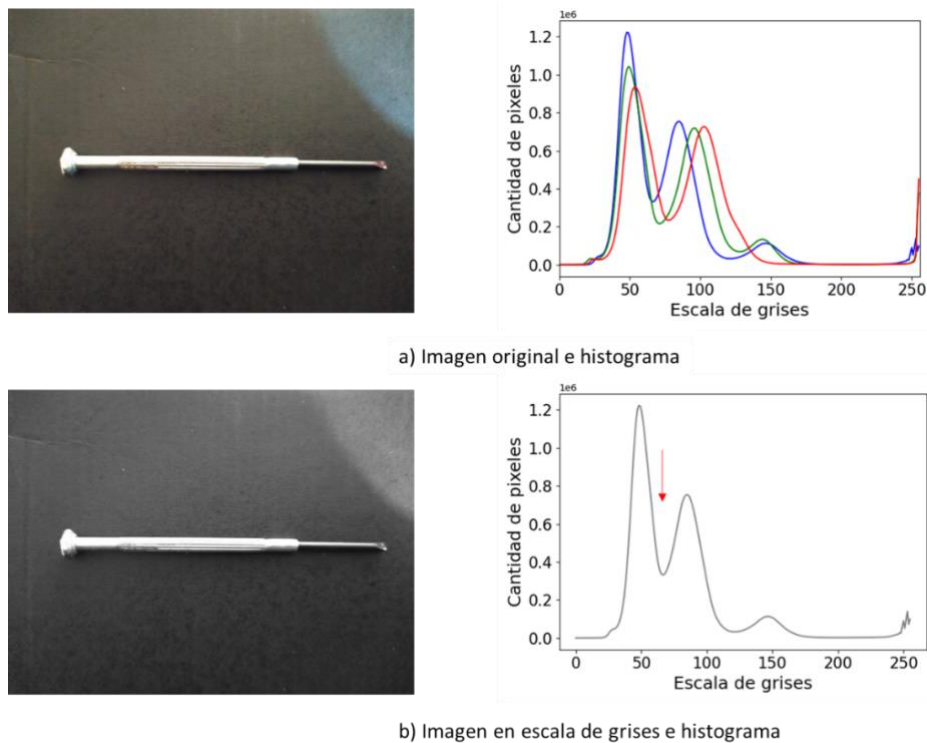
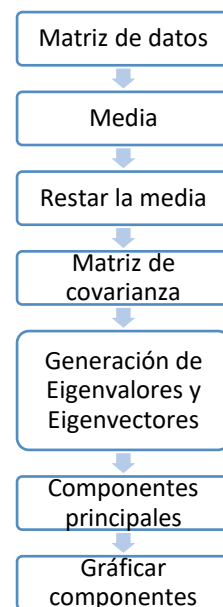


Figura 15. Ejemplo de histograma.

El proceso general del *PCA* se indica en la Figura 16. Primero se forma una matriz con los datos obtenidos de las imágenes, posteriormente se calcula la media con los datos correspondientes a sus dimensiones, una vez obtenido esto, se le resta la media a cada una de las filas y columnas de la matriz. Finalmente se calcula la matriz de covarianza, para generar los Eigenvalores y Eigenvectores para así obtener los componentes principales para proyectarlos mediante una gráfica. Las características principales de la imagen original se conservan después de aplicar el cálculo del algoritmo, para posteriormente, comparar el resultado con otras imágenes, puesto que sí coinciden, entonces la proyección generada corresponde a la imagen original, lo cual, puede ser comprobado con la graficación de la nueva imagen. También se compara de manera gráfica que la proyección (imagen), sea similar a la imagen original (Caballero Julián, Vidal Reyes, López Sánchez, y Jerónimo Ríos, 2017).



*Figura 16. Proceso de Análisis de Componentes Principales.*

Para desarrollar los algoritmos de identificación de patrones, es necesario considerar algunas variables como la iluminación por los cambios de apariencia que pueda ocurrir con respecto a las piezas u objetos a identificar; la posición de las piezas enfatizando en el ángulo en el que se encuentra porque puede dificultar la localización de la pieza de acuerdo con el algoritmo (Basulto Rodríguez, 2018).

El propósito del desarrollo del algoritmo de *PCA* como pretratamiento de imágenes en la RNC es el de mejorar el porcentaje de exactitud en clasificación. Para evaluar el rendimiento mediante métricas se tienen que utilizar ecuaciones como se menciona en la sección de Métricas de evaluación.

#### 4.5 Métricas de evaluación.

Menciona Tiwari (Tiwari, 2019), que para evaluar la calidad de una red neuronal, se pueden tomar en cuenta métricas como: Precisión, Sensibilidad, Exactitud y F1, los cuales, miden un aspecto en específico.

**Precisión:** Es la relación de las medias con la cantidad de datos totales. Por ejemplo, si se considera una carpeta con 50 imágenes a clasificar, pero el sistema solo clasificó 40, entonces la precisión es del 80 %, refiriéndose así al conteo de los verdaderos positivos totales. La ecuación para calcularlo es:

$$Precisión = \frac{VP}{VP+FP} \quad (6)$$

Donde:

*VP*: Verdadero Positivo

*FP*: Falso Positivo

**Exactitud:** la porción de imágenes en donde la etiqueta real coincide con la etiqueta de predicción. Es decir, si el sistema predice que la imagen corresponde a una pieza de metal con forma de círculo y el valor real indica lo mismo, se dice que hay exactitud. La ecuación para calcularlo es:

$$Exactitud = \frac{VP+VN}{VP+FP+FN+VN} \quad (7)$$

Donde:

*VN*: Verdadero Negativo

*FN*: Falso Negativo



**Sensibilidad o recuerdo:** la proporción de casos positivos reales que se identifican correctamente, sobre una porción de verdaderos positivos. Como ejemplo se considera que tan confiable es el modelo para realizar predicciones correctas. La ecuación para calcularlo es:

$$\text{Sensibilidad} = \frac{VP}{VP+FN} \quad (8)$$

**Puntuación F1:** Combina las métricas de precisión y sensibilidad. En caso de que se obtengan valores altos en ambas métricas, indica que el modelo detecta las clases y que el resultado es altamente confiable, además se evitan falsos negativos. En caso de tener la precisión con valores altos y la sensibilidad con valores bajos, indica que detecta la clase, pero el resultado es poco confiable. Si se tuviera un valor bajo en la métrica de precisión y alto en sensibilidad, indica que al modelo le cuesta trabajo detectar las clases, pero el resultado es confiable. Si las dos métricas tuvieran valores bajos, entonces los resultados no son aceptables.

$$F1 = 2 * \frac{(\text{Sensibilidad}) * (\text{Precisión})}{(\text{Sensibilidad}) + (\text{Precisión})} \quad (9)$$

Una herramienta muy útil para presentar los porcentajes de VP, FP, VN y FN de una manera más ordenada y que ayude a determinar mejor los resultados es la matriz de confusión. Las variables de la matriz de confusión ayudan a clasificar de manera óptima el resultado de los datos históricos, recabados de la ejecución del sistema de clasificación. Generalmente se utilizan grandes cantidades de datos a clasificar y para garantizar el resultado, se utiliza una matriz de confusión que tendrá la estructura como en la Figura 17 (Tiwari, 2019):

		Actual	
		VP	FP Tipo de error I
Pruebas	FN Tipo de error II	VN	

*Figura 17. Matriz de confusión.*

El porcentaje que se obtenga depende del tipo de métrica, puesto que, como cada métrica analiza un tipo de clasificación, puede resultar que para precisión se obtenga un 90 %, mientras que para sensibilidad un 80 %, esto quiere decir que el resultado de verdaderos positivos y negativos, respecto a las ecuaciones correspondientes, tiene un efecto sobre el resultado final, es así como se analiza si los parámetros a evaluar son los correctos y así mismo determinar la confiabilidad del modelo de red neuronal (Caballero Julián, Vidal Reyes, López Sánchez, y Jerónimo Ríos, 2017).

De acuerdo con el tipo de métrica (Precisión, Exactitud, F1 y Sensibilidad), en los algoritmos de Redes Neuronales Artificiales y Análisis de Componentes Principales, aplicados en el reconocimiento de objetos, que se encuentran en la literatura, se obtuvieron distintos porcentajes de evaluación de resultados, los cuales se presentan en la Tabla 5, refiriendo que para la clasificación de imágenes, el porcentaje que se maneja de Precisión va desde 97.1 % al 98.82 % y de exactitud del 92 % hasta 98.82 %. Los siguientes trabajos se consideran por el uso de redes neuronales artificiales para resolver determinados problemas relacionados a la clasificación de objetos o tratamiento de imágenes.

*Tabla 5. Recopilación de trabajos que usaron técnicas de Visión Artificial.*

Título	Aplicación	Método	Métrica	Porcentaje Obtenido
Reconocimiento de patrones en imágenes médicas por medio de redes neuronales convolucionales	Reconocimiento de imágenes (Resonancias magnéticas, rayos X, ultrasonido) para reconocer tejido tumoral.	Red Neuronal Convolutacional	Exactitud de la validación	92.0 % (N=100)
			Precisión	97.1 %

Título	Aplicación	Método	Métrica	Porcentaje Obtenido
(Santamaría Colula, 2019)			Exactitud final de la prueba	98.8 % (N=81)
Redes neuronales convolucionales para la clasificación de componentes independientes de rs-fMRI. (Mera Jiménez y Ochoa Gómez, 2021)	Análisis de imágenes de resonancia magnética funcional en estado de reposo.	Wavelet-PCA Red Neuronal: Perceptrón Multicapa (MLP)	Exactitud	98.82 %
Sistema de Reconocimiento de Patrones de Imágenes e Inferencia de Acciones de un Brazo Robótico en un Ambiente no Estructurado Aplicado al Proceso de Almacenaje Robotizado. (Morejón Ortega, 2019)	Control de un brazo robótico para realizar la clasificación de piezas.	Backpropagation Función de Objeto de Frontera Momentos invariantes	Efectividad	Con <i>backpropagation</i> : Mínima 90.69 %

### Tasa de error

Permite contabilizar los errores de clasificación que realiza la red neuronal y de esta manera determinar el rendimiento. Para determinarlo se hace uso de la ecuación (10), donde  $n$  pertenece al número de ejemplos del conjunto de datos sobre el que se evalúa el modelo. En algunos casos se refiere a la pérdida esperada, por lo tanto, la esperanza es igual a la tasa de error del clasificador (Berzal, 2018).

$$error = \frac{FP+FN}{n} = \frac{errores}{n} \quad (10)$$

En relación con la información recabada en el estado del arte, las Redes Neuronales Convolucionales son consideradas, como las más aptas para el análisis de datos a través de imágenes, destinadas para el reconocimiento facial o reconocimiento de objetos.

En una Red Neuronal Convolutiva existe una etapa donde se reduce la información de una imagen, distribuyéndola en diferentes capas y generando una respuesta rápida; sin embargo, existe la incertidumbre de pérdida de información puesto que si se discriminan

características importantes que constituyen la imagen de interés, difícilmente se le aportará una retroalimentación necesaria para el aprendizaje de la Red Neuronal Convolutiva, por lo tanto, se propone realizar el pretratamiento de las imágenes de entrenamiento para la RNC, con el algoritmo de Análisis de Componentes Principales, logrando así recopilar los descriptores y Eigenvalores de las imágenes, donde se extraen características necesarias reduciendo la pérdida de información.

La iluminación juega un rol relevante en los sistemas de reconocimiento, ya que son sensibles a la luz. Por lo que, es imprescindible realizar la prueba con diferentes niveles de iluminación. En ocasiones se tienen que restringir las condiciones como cantidad de iluminación, tipo de fondo para colocar piezas metálicas y tomar fotografía, aplicación de filtros para las imágenes, tomando en cuenta un intervalo de iluminación que sería la condición bajo la que trabajaría la red neuronal convolutiva en el lugar requerido.

En el desarrollo del proyecto es necesario involucrar etapas para realizar de manera efectiva el procedimiento de clasificación de acuerdo con los niveles de iluminación propuestos, el cual se expone a partir de la definición de la sección del Método.

## 5 MÉTODO

El desarrollo del proyecto se lleva a cabo en diferentes etapas (Figura 18), en el caso del algoritmo de la Red Neuronal Convolutiva con Análisis de Componentes Principales, el proceso comienza con la adquisición del *dataset*, sección en donde se realiza la normalización de imágenes. Posteriormente, se continúa con el diseño de experimentos (Diseño factorial  $3^4$ ), que indica el número de pruebas a realizar, luego se aplica el Algoritmo PCA con el cual se realiza el pretratamiento de imágenes, continua el entrenamiento de los datos con la RNC, sigue la validación del funcionamiento de la Red Neuronal Convolutiva y se concluye con la evaluación de los resultados. Para el caso del algoritmo de la Red Neuronal Convolutiva sin Análisis de Componentes Principales, se omite el paso de Algoritmo PCA.

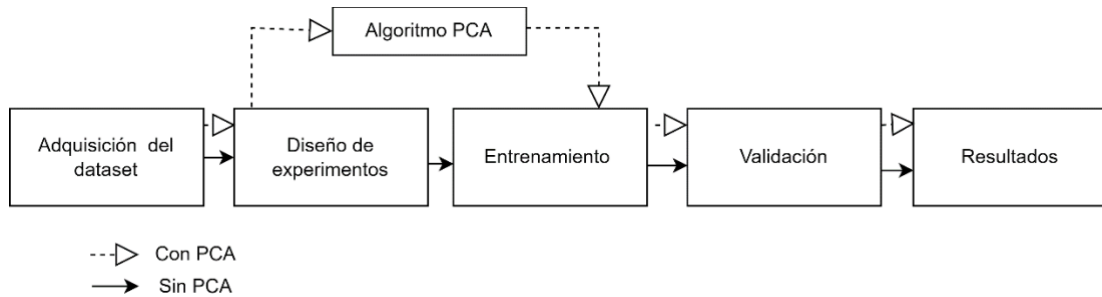


Figura 18. Etapas de experimentación

### 5.1 Diseño factorial $3^4$

Para realizar el proyecto se utiliza el diseño de experimentos factorial  $3^4$  (Figura 19). Como se tienen 4 variables y 3 niveles, corresponde a un total de 81 combinaciones, resultado de elevar  $3^4$ . El método se usa para determinar variables con mejor desempeño en la red neuronal convolutiva respecto a la métrica de exactitud. Las variables que se identifican como primordiales en el proyecto son: iluminación (1000 lx, 1500 lx y 2000 lx), cantidad de neuronas en la capa oculta (16, 32, 64), tipo de optimizador en la red neuronal (RMSProp, SGD y Adam) y número de componentes (2, 25, 50 componentes PCA). Un ejemplo de combinación es imágenes de 1000 lx, 16 neuronas en capa oculta, optimizador RMSProp y 2 componentes PCA; así como ese tipo de combinación, existen 80 combinaciones que se tienen que realizar. Las imágenes que se van a utilizar,

primeramente se les realiza un ajuste de iluminación para después generar las 81 combinaciones.

El experimento se realiza tres veces para determinar la veracidad de los resultados para el diseño factorial, puesto que se tienen 81 combinaciones por repetición, además de estar definido por el tiempo del desarrollo del proyecto.

En una primera repetición se evalúa que las 81 combinaciones de las variables se hayan sustituido dentro de la red neuronal, de lo contrario, se realiza combinación por combinación hasta completar la cantidad. Una vez completadas las combinaciones, se incrementa el valor de repetición de experimentación. El proceso se repite hasta obtener 3 repeticiones para evaluar el promedio y la desviación estándar de los porcentajes. Finalmente se obtiene la mejor combinación de acuerdo con el mejor rendimiento evaluado por la métrica de exactitud, tal como se propone en la sección de objetivos.

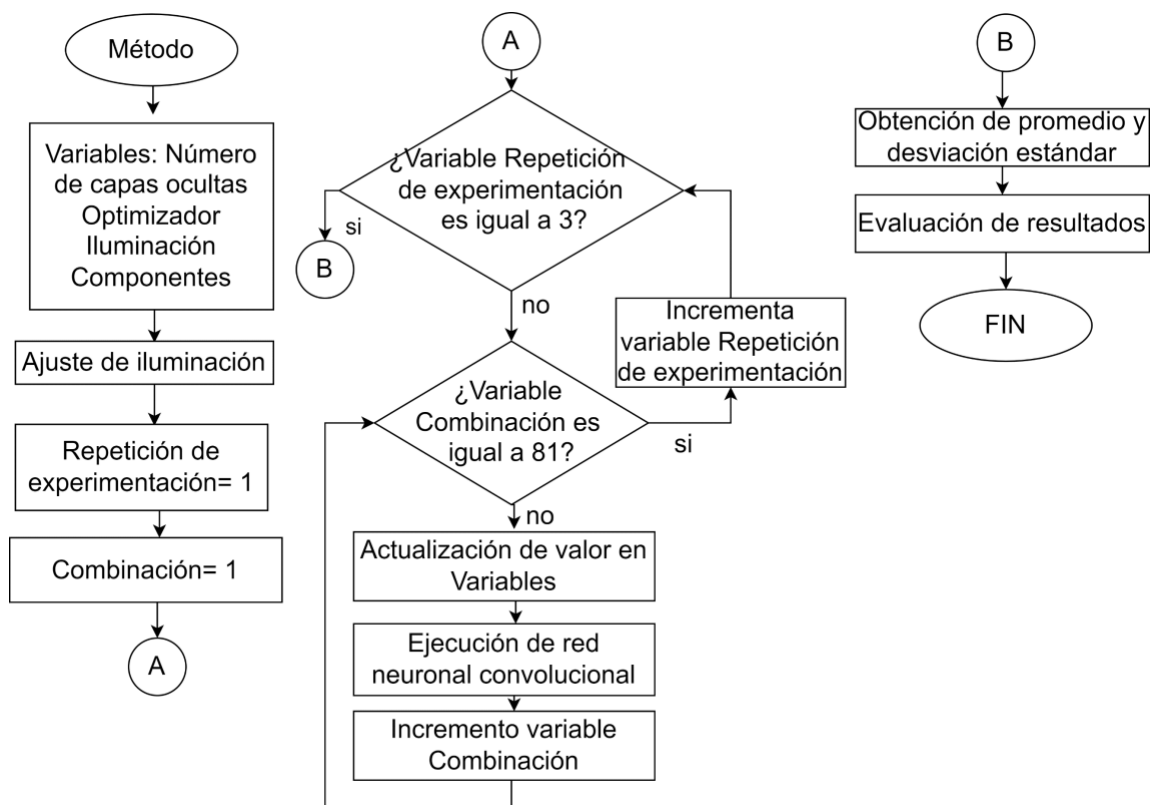


Figura 19. Etapas del diseño factorial

Las variables que intervienen en el proyecto son: número de componentes (representada por A), cantidad de iluminación (L) y las variables de neuronas en capa oculta y

optimizador se engloban en R, para que después de estimar los resultados del diseño factorial, se obtenga la combinación con el porcentaje de exactitud más alto, la cual indica los valores de las variables que se sustituyen en la red neuronal convolucional con y sin PCA que corresponde a la etapa de clasificación (C) y finalmente se evalúan los resultados para determinar de qué pieza se trata (T). La relación de los factores se representa en la Figura 20.

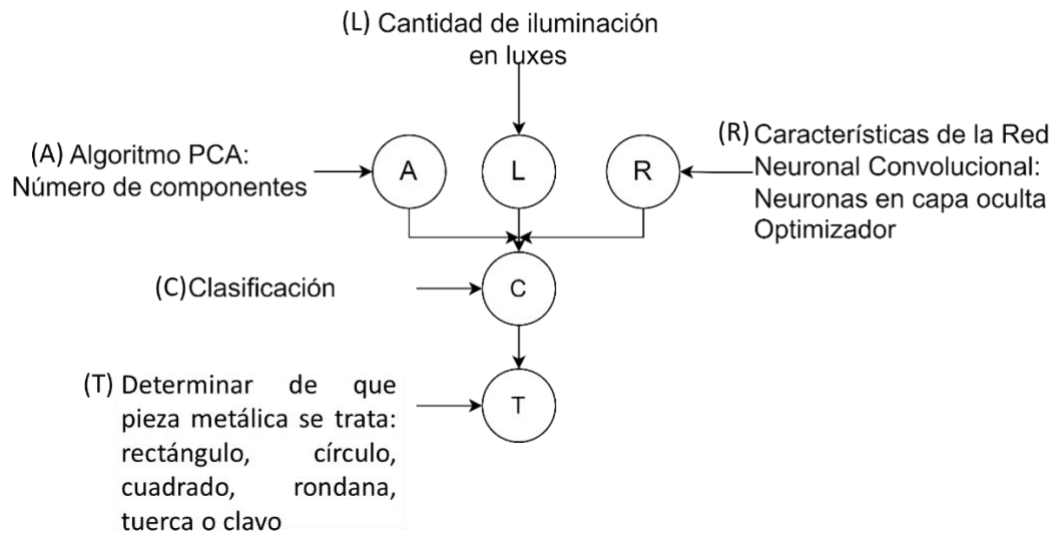
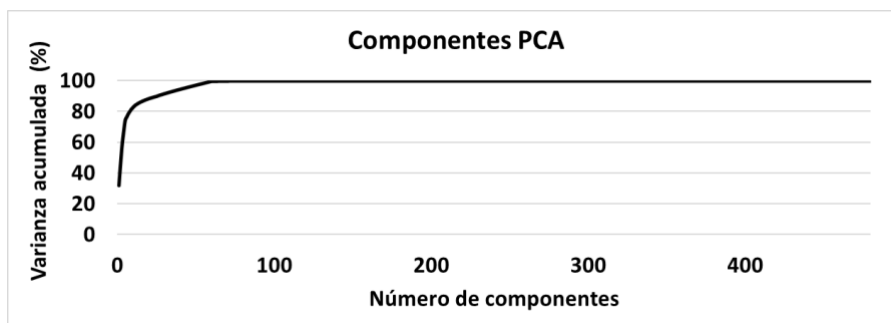


Figura 20. Relación entre factores.

A través del uso de PCA, se obtuvo que las características reducidas de la imagen comprenden desde 0 a 480 componentes (Figura 21). La cantidad de componentes se debe a la reducción de los Eigenvectores y Eigenvalores mediante el uso de la sentencia “`modelo_pca.components_`” de la librería Sklearn de Python, obteniendo así un primer proceso que es la obtención de la matriz de covarianza y la matriz de correlación. Como resultado se generó la varianza, que indica la cantidad de información contenida por cada uno de componentes y con la cual se puede recrear la imagen, seleccionando así tres cantidades de componentes que corresponde a: 2, 25 y 50. El criterio de selección es la cantidad de componentes teniendo en cuenta que la varianza acumulada debe proporcionar la mayor cantidad de información importante para las imágenes, por lo tanto se evalúa que con 2 componentes (valor bajo) se genera el 46.05 % de varianza acumulada, con 25 componentes (valor medio) el 90 % de varianza acumulada y con 50 componentes (valor alto) el 97.25 % de varianza acumulada.



*Figura 21. Comportamiento de la varianza acumulada en función del número de componentes*

La letra L simboliza la iluminación: 1000, 1500, 2000 lx. La variable iluminación influye en las imágenes definiendo la opacidad de estas. Los niveles de iluminación están definidos de acuerdo con la Norma Oficial Mexicana NOM-025-STPS-2008, que indica que el mínimo de iluminación en un entorno de maquinado es de 1500 lx, se consideran dos niveles más de iluminación para determinar cuál es más adecuado siendo 1000 lx como nivel bajo y 2000 lx como nivel alto.

La letra R hace referencia a las características que involucran a la red neuronal a través del uso de optimizadores. En pruebas rápidas se evaluaron los optimizadores Adagrad, Nadam y Adadelta, Adam, RMSProp y SGD a través de la matriz de confusión y se determinó que proporcionan mejores resultados de clasificación los tres últimos. En la Figura 22 se muestran los resultados obtenidos de las pruebas rápidas con los optimizadores SGD, Adam, RMSProp, Nadam y Adadelta. La ejecución se realizó con la red neuronal convolucional con PCA, con 25 componentes, 64 neuronas en capa oculta, imágenes de cámara de 1.3 MP expuestas a 1500 lx. Por lo tanto, los optimizadores con mejor desempeño son SGD con 98.33 % de exactitud, Adam con 96.66 % de exactitud y RMSProp con 95.83 % de exactitud, mientras que con Nadam se obtuvo 91.6 % de exactitud y Adadelta 87.5 % de exactitud, entonces, los dos últimos optimizadores no cumplen con el valor de la hipótesis (93 % de exactitud), y se opta por evaluar el valor de los optimizadores Adam, SGD y RMSProp, que, como se expone en la Figura 22, el mejor resultado fue para SGD:



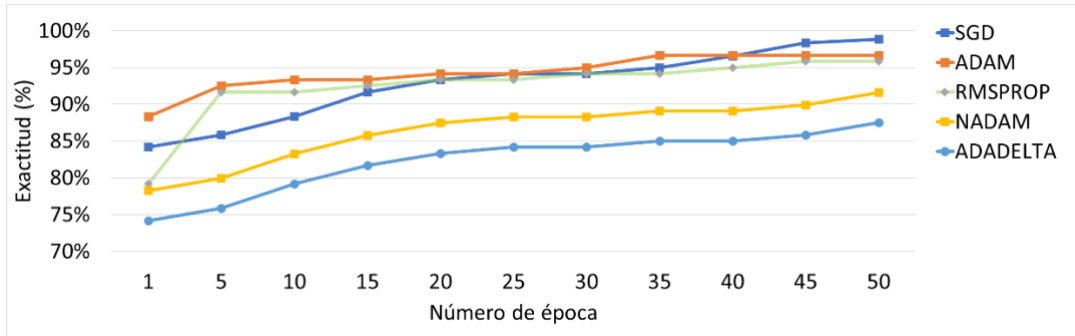


Figura 22. Gráficas del promedio de exactitud para los optimizadores SGD, Adam, RMSProp, Nadam y Adadelata

Para el caso de las capas ocultas, no hay una ecuación que indique la cantidad total de neuronas exactas (Gazzaz, Yusoff, Aris, Juahir, y Ramli, 2019), por lo que se propone la experimentación con 16, 32, 64 e indicados como niveles bajo, medio y alto en experimentación, siendo 64 un nivel alto porque si se utilizan más neuronas ocasiona un sobre entrenamiento puesto que se alcanza el 90 % de exactitud en las primeras épocas y posteriormente el porcentaje disminuye; si fuera menos de 16 neuronas, existe bajo entrenamiento porque no permite que la red neuronal se ajuste de acuerdo a los parámetros establecidos y la exactitud no mejora a través de las épocas (Martín, López, y Martín, 2006), además en el entrenamiento y validación con el uso de 32 neuronas proporcionó un porcentaje alrededor del 90 % de exactitud, tal como se expone en la sección de resultados del diseño factorial (Figura 58-Figura 60).

La relación de todas las variables se expresa matemáticamente como  $p(C|A, L, R)$ , y la combinación de las variables dará como resultado la clasificación de la imagen (denotada por la letra C). Finalmente, el resultado (T) será el tipo de pieza metálica clasificada como regular: cuadrado, rectángulo, círculo, rondana o irregular: tuerca, clavo y se expresa este resultado como  $p(T|C)$ .

A partir de las variables definidas se puede construir la matriz de experimentación mostrada en la Tabla 6.

Tabla 6. Matriz de experimentación.

Número de neuronas en capa oculta	Optimizador	Iluminación	Componentes
X=16	A=RmsProp	1=1000lx	C1=2
Y=32	B=Adam	2=1500lx	C2=25
Z=64	C=SGD	3=2000lx	C3=50

Los valores que pueden tomar las variables del diseño factorial  $3^4$  se presentan en la Tabla 6. Las combinaciones resultantes se muestran en la Tabla 7.

Tabla 7. Ejemplo de combinaciones de matriz de experimentos

Numero de neuronas en capas oculta	Optimizador	Iluminación (lx)	Componentes
16	SGD	1500	50
64	RmsProp	1500	2
16	RmsProp	1500	25
16	RmsProp	1000	50
16	SGD	2000	2
16	SGD	1000	50
32	RmsProp	1500	50
32	SGD	1000	2

El número de combinaciones de  $3^4$  es de 81, para cada combinación se genera un experimento distinto y se evalúa con la métrica de exactitud, para que al final se determine la mejor combinación para la clasificación de imágenes. El total de combinaciones se encuentra en el apartado de resultados del diseño factorial incluido en la sección de Anexos.

## 5.2 Proceso general del algoritmo

Como primer paso (Figura 23) se carga el *dataset* de entrenamiento para obtener las etiquetas respectivas: círculo, cuadrado, rectángulo, rondana, clavo, tuerca. Las etiquetas se obtienen de manera automática, a través de la lectura del directorio de las imágenes de entrenamiento que contiene las carpetas de cada una de las clases. El nombre de las carpetas se coloca de forma manual, sin embargo, el nombre de las imágenes que contiene

cada carpeta es distinto, puesto que se otorga un nombre diferente al momento de almacenar las imágenes, las imágenes contienen un rango de píxeles de 0 a 255 y se estandarizan los datos en un rango entre 0 y 1.

Con los componentes de las imágenes, tales como líneas y curvas, se evita la información redundante ya que puede ocasionar que la clasificación no sea la correcta, por lo tanto, como primer paso en la aplicación con el análisis de componentes, se obtiene la media de los datos para luego calcular la matriz de covarianza, tanto de entrenamiento, como de validación; multiplicada por la matriz que resulta de la multiplicación de la matriz original con la matriz unitaria. El resultado genera valores positivos y negativos que indican el tipo de relación ya sea inversa que son los valores menores a la media o directa que son los valores mayores a la media, entonces, la relación entre variables sirve para descartar variables que no tengan información directa o importante.

A continuación, se calculan los Eigenvalores y los Eigenvectores, mediante el resultado de los componentes, se generan los vectores de entrenamiento y de validación que servirán para la red neuronal convolucional, permitiendo así la obtención de los pesos adecuados. Para comprobar el funcionamiento se hace la lectura de datos de la carpeta de validación con la que se realiza la predicción de imágenes, este proceso consiste en asignar la etiqueta correspondiente a cada una de las imágenes. Con los resultados que se generen históricamente de la clasificación de todas las imágenes disponibles en la carpeta, se obtiene la matriz de confusión para evaluar las métricas precisión, sensibilidad, F1 y exactitud. Esta última es la de mayor interés porque evalúa el acercamiento del valor medido y el valor real de las imágenes a clasificar, la cual se verifica que corresponda a un porcentaje por lo menos del 93 % de lo contrario, se realizan los ajustes correspondientes en la etapa de la obtención de los Eigenvalores y Eigenvectores (el proceso se presenta en el diagrama de la Figura 23).

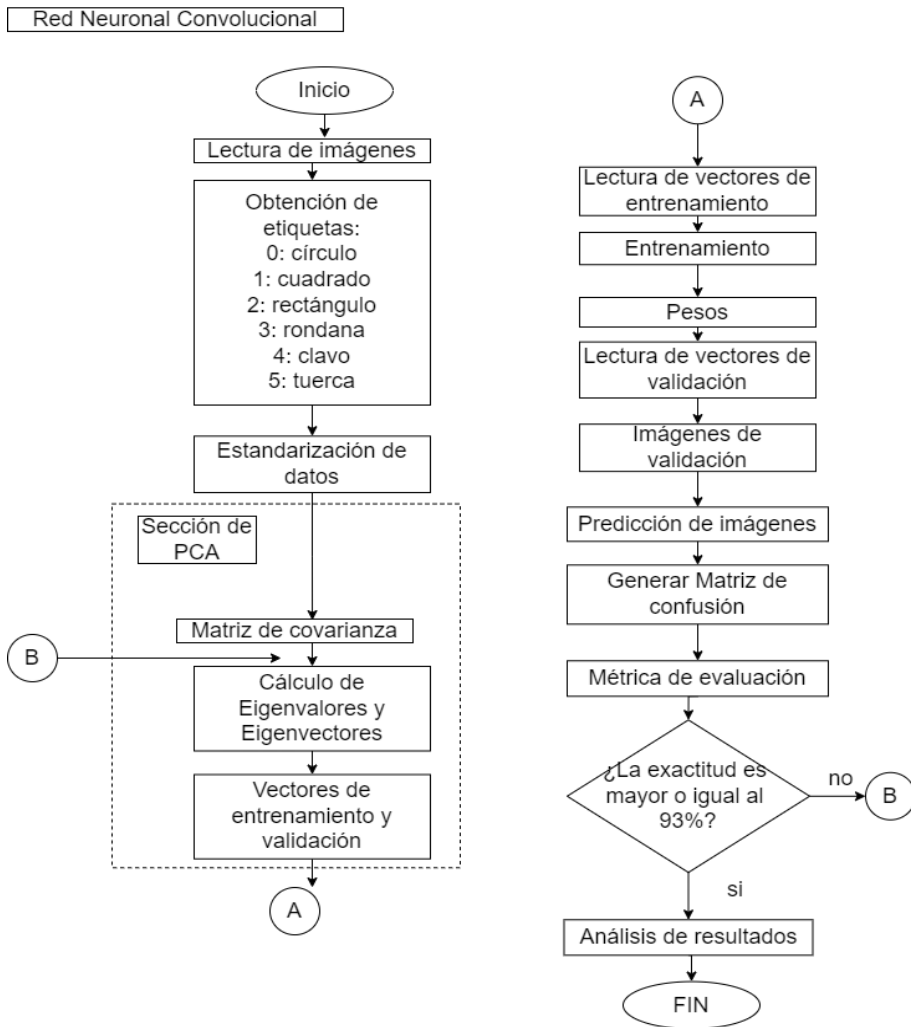


Figura 23. Etapas del proceso de PCA en RNC

### 5.3 Requerimientos o especificaciones del sistema

Los requerimientos para llevar a cabo la elaboración del proyecto se contemplan de acuerdo con los siguientes aspectos:

#### Ambiente físico

Es conveniente que se tenga la posibilidad de poder controlar las entradas de iluminación (ventanas, lámparas, entre otros), con el fin de realizar pruebas con las variaciones de iluminación los más exactas posibles.

#### Usuarios y factores humanos

Etapa de clasificación de piezas de maquinado. El sistema podrá ser utilizado por la persona que tenga conocimiento del funcionamiento del sistema y que requiera como resultado que las piezas metálicas correspondan a piezas regulares (cuadrado, círculo, rectángulo y rondana) o irregulares (clavo y tuerca).

### Funcionalidad del sistema

Tabla 8. Características de las cámaras de 1.3 MP y 8MP.

Cámaras		
Características	1.3 MP (European machine vision association, 2006)	8 MP (Xatakandroid, 2019)
Modelo	Basler acA1300-200um	Xt1601
Sensor:	Python 1300	BSI (Black Side Illumination)
Tipo de sensor:	CMOS	CMOS
Resolución	1280 x 1024 pixeles	720 x 1280 pixeles
Tipo de interface	USB 3	Servidor DroidCamX

1 lámpara incandescente de 100W.

1 lámpara de luz blanca, 75 W

- Uso de un dimmer sencillo de botón:
  - Marca: Supplier
  - Modelo SD-305
  - Voltaje: 127V

Para realizar las mediciones del nivel de iluminación, se requiere de un luxómetro “UNIT 382 Series”. Las características son las siguientes (Ingprosuppliers, 2021):

- Mediciones en LUX: 20, 200, 2000, 20000 lux
- Mediciones en CD: 2, 20, 200, 2000 CD

### Restricciones:

El rango mayor del nivel de iluminación es de 2000 lx y el rango menor es de 1000 lx. También se toma en cuenta la Norma Oficial Mexicana *NOM-025-STPS-2008*, que establece que para entornos de maquinado y de inspección de piezas se utilizan

aproximadamente 1000 lx. En entornos para realizar maquinado donde se requiera de una mayor precisión y procesos de ensamble, se usan 1500lx. Mientras tanto, para interiores se utilizan aproximadamente 100 lx. (Diario Oficial de la Federación, 2008)

## 5.4 Diseño e implementación

El desarrollo del proyecto se llevó a cabo en diferentes etapas, en las cuales se definen las piezas y parámetros a utilizar. A continuación, se presentan cada una de las etapas.

### 5.4.1 Selección de piezas metálicas

Para realizar la clasificación de piezas metálicas, se usan específicamente las piezas: tuerca, clavo, rondana, círculo, rectángulo, cuadrado, que son representativas a las formas fáciles de conseguir, económicas y comúnmente fabricadas o terminadas con procesos de maquinado, también se suelen utilizar procesos para su embalaje. Las piezas metálicas se eligieron de acuerdo con las características de dos tipos de clases: piezas regulares (Figura 24) e irregulares (Figura 25). La aplicación de cada una de las piezas es base para crear otros tipos, por lo que es esencial considerar primero las formas básicas e ir identificando la dificultad que se tiene para clasificarlas.



Figura 24. Piezas metálicas regulares



Figura 25. Piezas metálicas irregulares

### 5.4.2 Establecimiento de las condiciones de iluminación

Para variar los niveles de iluminación de la lámpara incandescente de 100 W, se conectó un *dimmer* que contiene una perilla en su diseño, tal como se muestra en el diagrama de la Figura 26.

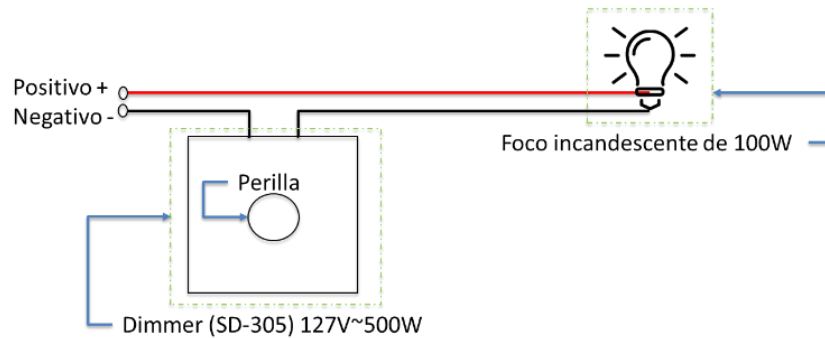


Figura 26. Diagrama eléctrico de conexión dimmer-lámpara

En relación con la lámpara que proporciona luz blanca, se optó por aumentar y disminuir la distancia de la lámpara a la mesa de trabajo para conseguir los niveles de iluminación propuestos en cantidad de *luxes* (Figura 27). En ambos casos se midieron los niveles a través del luxómetro UT382 series.

Para la captura de imágenes, la cámara se coloca a 20 cm de la pieza para delimitar el área y se utiliza un fondo negro sobre el cual se coloca la pieza de trabajo. Respecto a la iluminación, la lámpara (puede ser reemplazada por lámpara con iluminación fría y cálida) está conectada a un cable para ser movable y así obtener los niveles de *luxes* propuestos mediante la variación con el *dimmer*. La lámpara está colocada a una distancia de 40 cm. El luxómetro se coloca de tal forma que permita medir la cantidad de *luxes* que llegan hasta donde está ubicada la pieza.

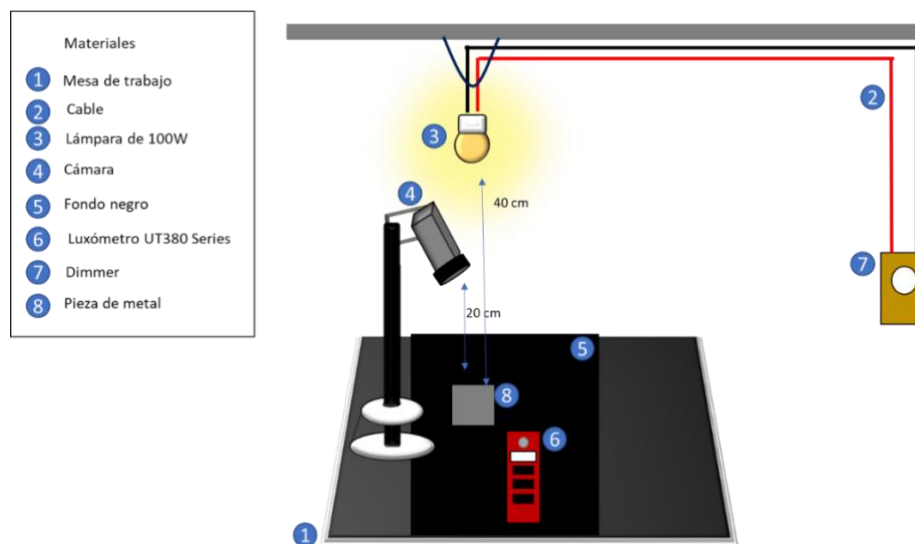


Figura 27. Entorno de trabajo.

La variación de iluminación permite la obtención de 1000, 1500 y 2000 lx que son los que se requieren para realizar las experimentaciones.

### 5.4.3 Captura de imágenes

Para la captura de las imágenes se diseñó un módulo que detecta los contornos de las imágenes de las piezas metálicas con el filtro de *Canny* y posteriormente binarizar la imagen para realizar el recorte mediante el algoritmo de Gauss, ya que permite filtrar el ruido (diferentes contrastes entre píxeles, ruido proveniente de la cámara) que se presente en la imagen, a partir del uso de las cámaras de 1.3 MP y de 8 MP.

En las imágenes tenues con escala de 1000 a 1500 *luxes*, se crea un filtro de tamaño 7 x 7 píxeles para la aplicación de filtro *Canny* y obtener el valor de umbral de la imagen. Por otro lado, con el fin de eliminar el ruido en la imagen y normalizarla a través del filtrado, se utiliza el filtro de *Gauss*, también con tamaño de 7 x 7 píxeles.

Además del ruido filtrado con *Canny* y *Gauss*, también se considera como ruido a aquellos elementos dentro de la imagen que no son parte del objeto de interés, por lo tanto, es necesario extraer el objeto y descartar el resto de la imagen capturada, a través de la identificación de los 4 bordes que la conforman: superior, inferior, izquierdo y derecho.

Para determinar cada uno de los bordes, se emplearon 4 matrices de tamaño 3x3 para guardar las coordenadas (fila, columna) y un valor de control de cada borde: Una primera matriz para guardar los valores del borde superior (todos los valores inicializados en 0), otro para el borde inferior (todos los valores inicializados en 0) y otra matriz para guardar el borde de lado izquierdo (valor 1 y 2 inicializados en 0 mientras que el valor de control inicializado en valor del ancho de la imagen). Para el caso del algoritmo sin PCA, se omite la sección de Algoritmo PCA, denotado por línea punteada y continua con la etapa de entrenamiento. El proceso se muestra en la Figura 28.



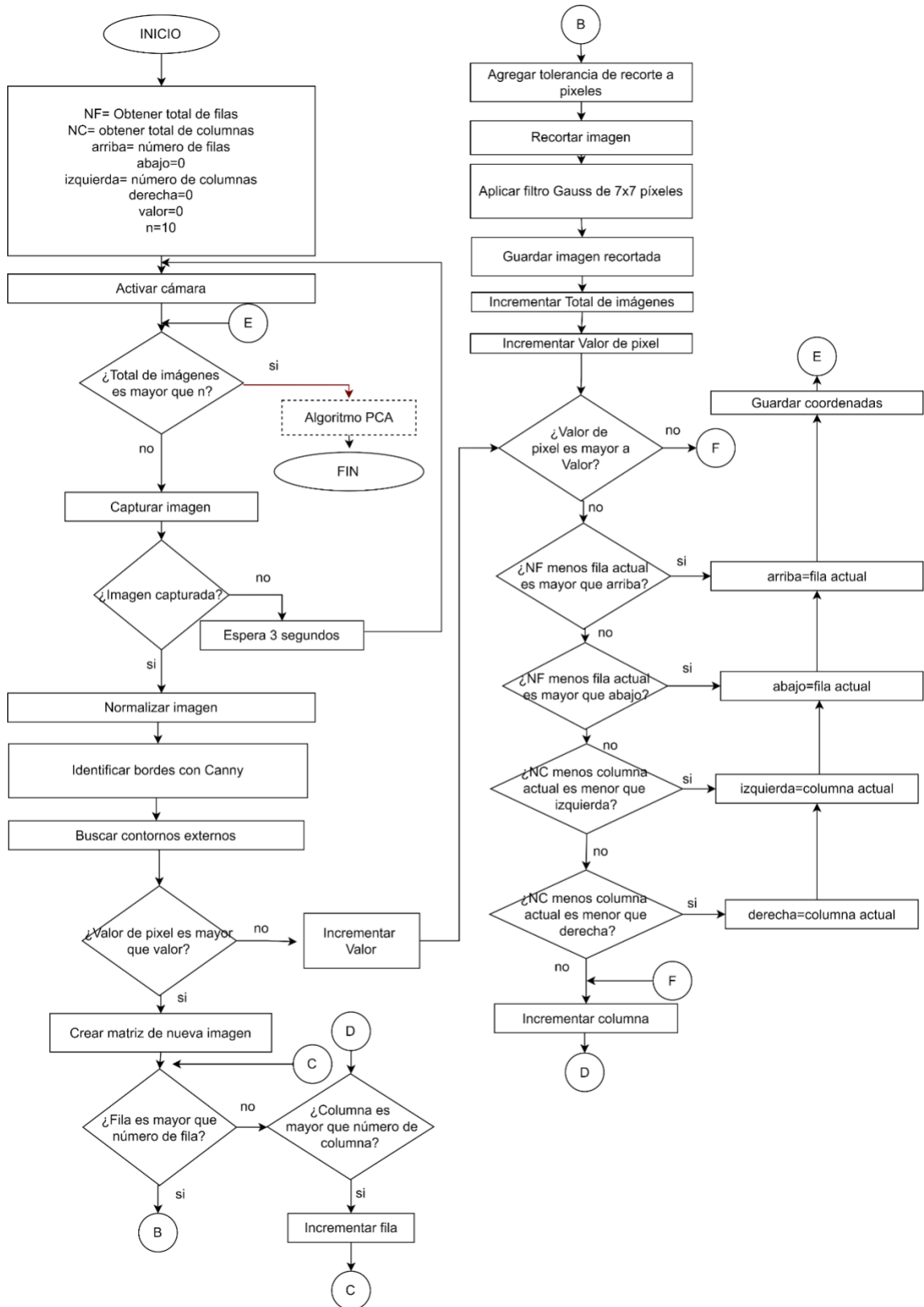


Figura 28. Algoritmo para captura y recorte de imágenes.

A partir del algoritmo de captura y recorte de imágenes, el total de imágenes que se obtuvo es de 1080, las cuales fueron obtenidas a partir de lo que se indica en Tabla 9-9, respecto a las dos cámaras puesto que se experimentó con dos lámparas una que proporciona luz fría y otra luz cálida, además se considera la división del *dataset* como 60 % de imágenes para entrenamiento, 40 % para validación.

Finalmente, la carpeta de entrenamiento tiene 6 clases: cuadrado, rectángulo, círculo, rondana, clavo, tuerca. Cada una de las carpetas tiene 120 imágenes, destinadas para el entrenamiento de la red neuronal convolucional (piezas regulares e irregulares, Tabla 9). Para realizar la validación (Tabla 10), se tomaron 360 imágenes.

*Tabla 9. Imágenes por pieza para entrenamiento*

Rangos de iluminación (lx)	Cantidad de imágenes por pieza					
	Cuadrado	Círculo	Rectángulo	Rondana	Clavo	Tuerca
<b>2000</b>	40	40	40	40	40	40
<b>1500</b>	40	40	40	40	40	40
<b>1000</b>	40	40	40	40	40	40
<b>Total de imágenes: 720</b>	120	120	120	120	120	120

*Tabla 10. Imágenes por pieza para validación*

Rangos de iluminación (lx)	Cantidad de imágenes por pieza					
	Cuadrado	Círculo	Rectángulo	Rondana	Clavo	Tuerca
<b>2000</b>	20	20	20	20	20	20
<b>1500</b>	20	20	20	20	20	20
<b>1000</b>	20	20	20	20	20	20
<b>Total de imágenes: 360</b>	60	60	60	60	60	60

Las imágenes al ser capturadas mediante las cámaras de 1.3 MP y de 8 MP, tienen características diferentes de acuerdo con la relación aspecto, algunas imágenes tienen tamaño de 340 x 220 píxeles, esto dificulta que la red neuronal analice las imágenes de manera correcta, por lo tanto, es conveniente normalizar este tamaño a 150 x 150 píxeles para redimensionar el tamaño de las imágenes y reducir la cantidad de datos de entrada a la red neuronal. El algoritmo también determina qué imágenes no necesitan el balance y se siguen considerando como parte del *dataset* sin aplicarles balance de iluminación, esto evita que estén saturadas o con luminosidad alta (determinado por el cálculo de los

percentiles, ecuación 1), entonces solamente se realiza el tratamiento de las imágenes que lo requieren.

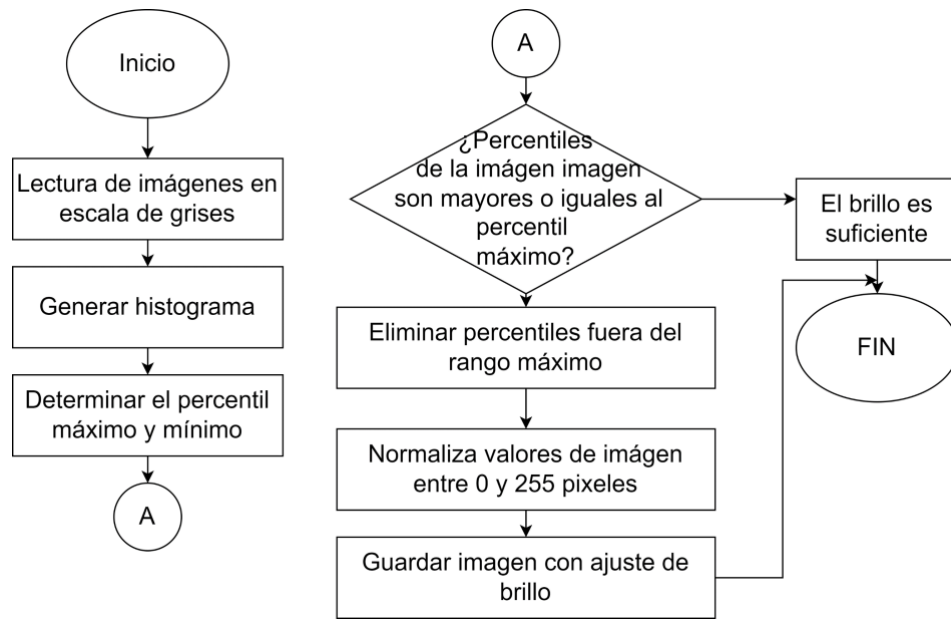


Figura 29. Balance de iluminación

#### 5.4.4 Tratamiento de las imágenes

A las seis clases se les aplicó el algoritmo de Análisis de Componentes Principales (PCA, Figura 30). Empieza por la captura de la imagen, la obtención del arreglo y posteriormente aplicar la normalización de los pixeles, después se obtiene la matriz de covarianza y finalmente una nueva proyección que es la graficación de la imagen.

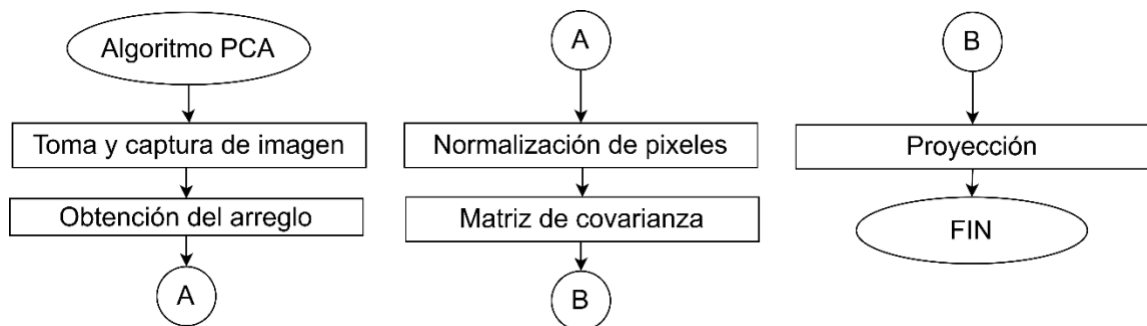


Figura 30. Pasos para el desarrollo del proceso de Análisis de Componentes Principales.

Después de la obtención del *dataset* y del tratamiento de las imágenes con el uso de PCA, se utiliza el modelo de la red neuronal para realizar la clasificación de las piezas metálicas.

#### 5.4.5 Diseño y codificación de la red neuronal

Para el desarrollo del presente trabajo se utiliza Python®, por ser un lenguaje multiplataforma (Mac, Linux, Windows), que cuenta con librerías dedicadas, así como herramientas para el desarrollo de redes neuronales artificiales, incluidas las convolucionales, además de ser un software libre.

Una de las versiones más actuales de Python es la 3.X (haciendo referencia a la X a alguna versión como 6, 7, 8, 9), las cuales son las más utilizadas para desarrollo de proyectos de inteligencia y visión artificial en la actualidad debido a la compatibilidad con las librerías siguientes: *TensorFlow*, *PyTorch*, *Keras*.

De las tres librerías se destaca *Keras* (compatible con la versión 3.6 de Python) que ofrece la posibilidad de poder diseñar un modelo propio de la red neuronal artificial. Como gestor de trabajo se utiliza *Google Colab* para trabajar con GPUs (Unidad de Procesamiento Gráfico), con almacenamiento en la nube.

El esquema de la red neuronal convolucional se muestra en la Figura 31. Las etapas con PCA pertenecen a las secciones indicadas con una flecha de color azul, mientras que las etapas sin PCA se muestran con flechas de color rojo. El total de datos de entrada para la red neuronal convolucional está conformado por el conjunto de imágenes del *dataset*. El tamaño de la matriz se obtiene a través de la función de Python *train\_test\_split*, que determina la división del *dataset* para entrenamiento y validación, por lo tanto, se crea una matriz de tamaño de 340 x 220 píxeles. En la red neuronal convolucional, los datos son multiplicados por los pesos correspondientes y procesados en la capa oculta donde intervienen los valores de experimentación correspondientes al número de neuronas ya sean 16, 32 o 64, luego, cada uno de los resultados se suman para conformar una sola salida con la cual se identifica una clase.

Para que la red neuronal convolucional genere un resultado, debe recibir una señal producida con la función de activación, en la RNC se utiliza *Softmax*. La función de activación *Softmax* va a generar la sumatoria de los resultados por datos dentro de un rango

entre 0 y 1, de tal forma que todos los datos sumen 1 y posteriormente se calcula la probabilidad que exista para pertenecer a una clase, correspondiendo a 0: rectángulo, 1: cuadrado, 2: círculo, 3: rondana, 4: clavo, 5: tuerca.

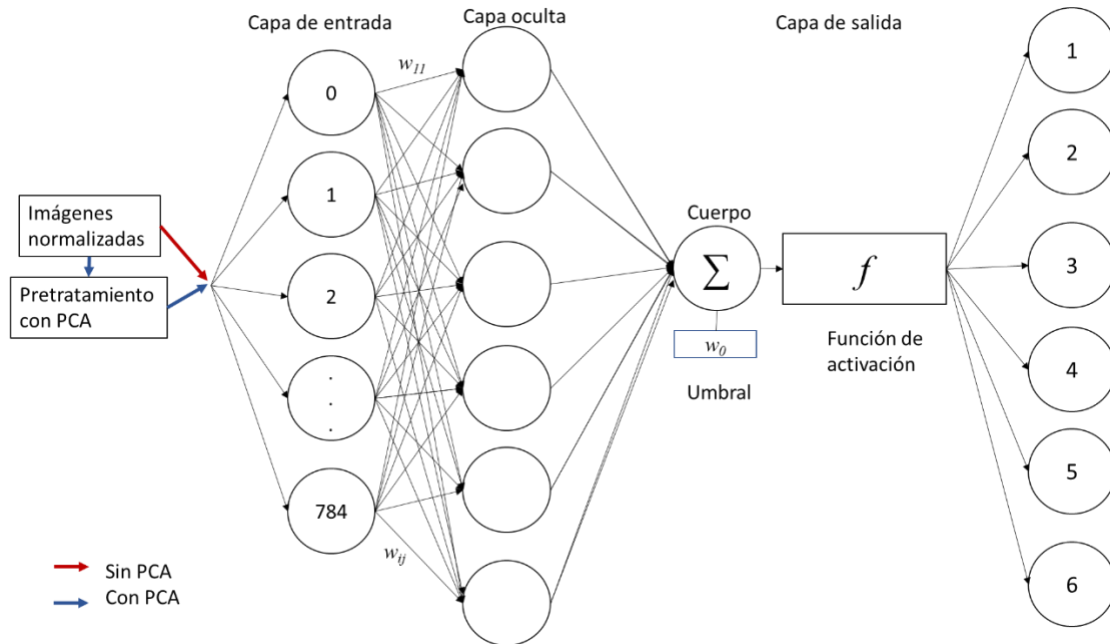


Figura 31. Red Neuronal Convencional

#### 5.4.6 Etapa de entrenamiento de la red neuronal

Se utilizan las librerías de *TensorFlow* y *Keras* para la red neuronal convolucional en el lenguaje Python y en cuanto a la ejecución del código se utiliza *Google Colab*; así como también *Numpy* para realizar operaciones con arreglos, *Os* para leer los directorios de las imágenes y *Matplotlib* para graficar.

La red neuronal convolucional (Figura 32) cuenta con los parámetros:

**Capa de entrada:** se realiza el pre-procesamiento de la imagen para la entrada de datos de la red neuronal convolucional. Se reciben las imágenes de un tamaño de 150 x150 píxeles con solo una dimensión de profundidad, debido a que se encuentran en escala de grises. Durante esta etapa se realiza la rotación y un muestreo de la imagen se obtienen fragmentos o secciones de la imagen original, con el fin de obtener de una imagen varias otras en diferentes posiciones considerando las posibles variaciones de las que se puede encontrar al momento de realizar el reconocimiento de las piezas.

Se establece el proceso secuencial para la red neuronal y a continuación se definen los parámetros para las capas. En la capa de entrada se recibe el tamaño de cada una de las imágenes que es 150 x 150 píxeles, se definen cuatro neuronas mediante una matriz de tamaño 3x3 píxeles. Se realiza el proceso de desconexión (*dropout* = 0.5) con un 50 % y para filtrar las características de las imágenes se utiliza *MaxPooling*, para que a través de filtros más pequeños de tamaño 2x2 píxeles se vaya recorriendo la imagen, recabando así una nueva matriz.

Capa para extraer características: al conjunto de imágenes recibidas en la capa de entrada se les aplica un proceso de filtrado, para obtener diferentes tamaños y secciones de la imagen, para descomponerla en otras imágenes. El primer paso es obtener la matriz de covarianza para poder realizar el cálculo de Eigenvalores y Eigenvectores que sirve para obtener la varianza de cada uno de los componentes de la imagen, por lo tanto, si no se obtienen no se podrá calcular la varianza la cual contiene el valor promedio y las diferencias entre cada componente, estimando el valor para poder proyectar las imágenes con otra dimensión más pequeña a la de entrada que era de 784 componentes. Después se definen la cantidad de neuronas (16, 32 o 64), tamaño de filtro (4 x 4 píxeles) y función de activación (*Softmax*). Se realiza nuevamente *MaxPooling* para filtrar la imagen nuevamente y se genera otra imagen.

Finalmente, en la capa de salida se colocan los parámetros donde se reciben las neuronas (25 que recibieron inicialmente), función de activación *Rectified Lineal Unit (ReLU)*, *Dropout* del 50 % y se determina cuantas clases se van a generar al final. Se emplean dos funciones de activación porque cada una de las capas debe contenerla para poder realizar el procesamiento de la imagen y finalmente la clasificación.

Con las características de la capa de entrada, capa oculta y capa de salida, se realiza el entrenamiento de la red neuronal y para comprobar el funcionamiento de la red, se utilizan las imágenes de validación para clasificarlas y evaluar los resultados a través de la matriz de confusión.

Para el caso de la red neuronal convolucional, se omiten los pasos que se indican en el diagrama con línea punteada, que corresponden a: crear matriz de covarianza, generar lista de Eigenvalores y Eigenvectores, sumar Eigenvalores, calcular varianza acumulada y

varianza explicada, aplicar algoritmo PCA. Los parámetros que no se omiten son los mismos en ambos casos.

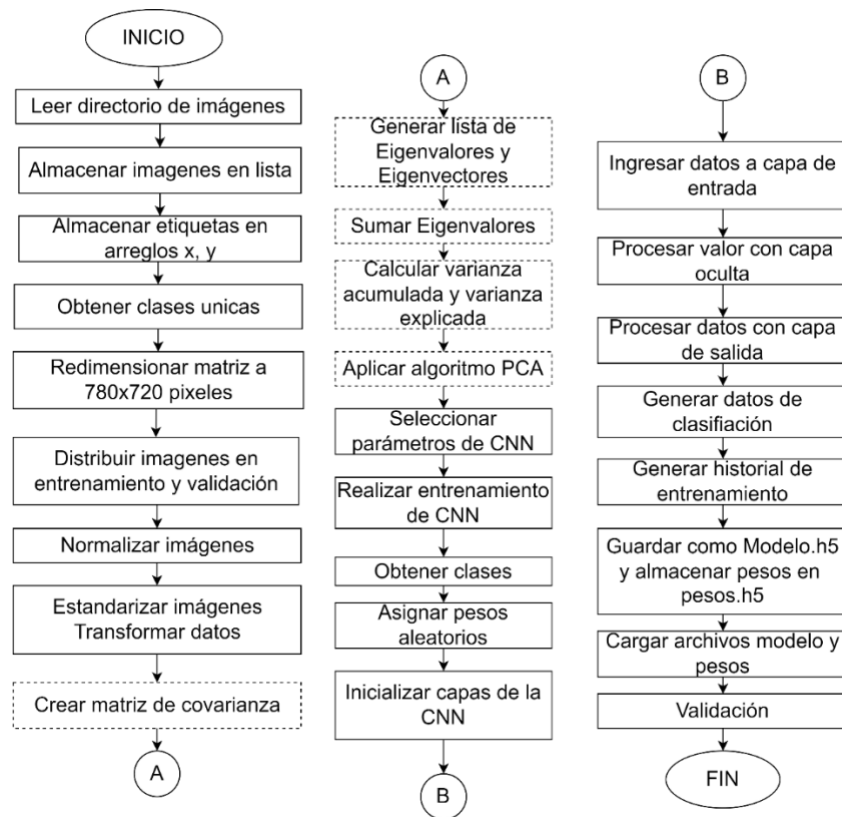


Figura 32. Proceso de entrenamiento de la Red Neuronal Convolutacional con y sin PCA

#### 5.4.7 Etapa de validación de Red Neuronal Convolutacional

Para validar los datos de la red neuronal se utilizaron las librerías *sklearn.metrics*, que proporciona la herramienta necesaria para medir la clasificación de las imágenes a través de la matriz de confusión (Figura 33).

Primero se leen las etiquetas de entrenamiento y de validación para almacenarlas en un respectivo arreglo y obtener la cantidad de etiquetas para saber si se han comparado los datos de ambos arreglos. El resultado de la comparación puede catalogarse en algún caso siguiente, ya sea en FN (Falsos negativos) que ocurre cuando el resultado real es positivo y la predicción es negativa, FP (Falsos Positivos) si la predicción es negativa y el resultado es positivo, VN (Verdaderos Negativos) cuando el valor real y la predicción son negativos; en cada caso se almacena el valor obtenido. Si el valor real y la predicción son positivos,

se trata de un VP (Verdadero Positivo) y de igual manera se almacena el valor en una variable.

Después se obtiene el total de cada una de las variables generadas, pues con los resultados se van a poder calcular las métricas a través del uso de la librería de *sklearn.metrics*. Mediante *classification\_report*, se crean los resultados correspondientes a sensibilidad, precisión, exactitud y F1.

Visualmente se muestran las clasificaciones de acuerdo con las clases utilizadas en una matriz que tiene la distribución de los puntajes de las etiquetas, de esta manera, se pueden evaluar de mejor manera los datos clasificados.

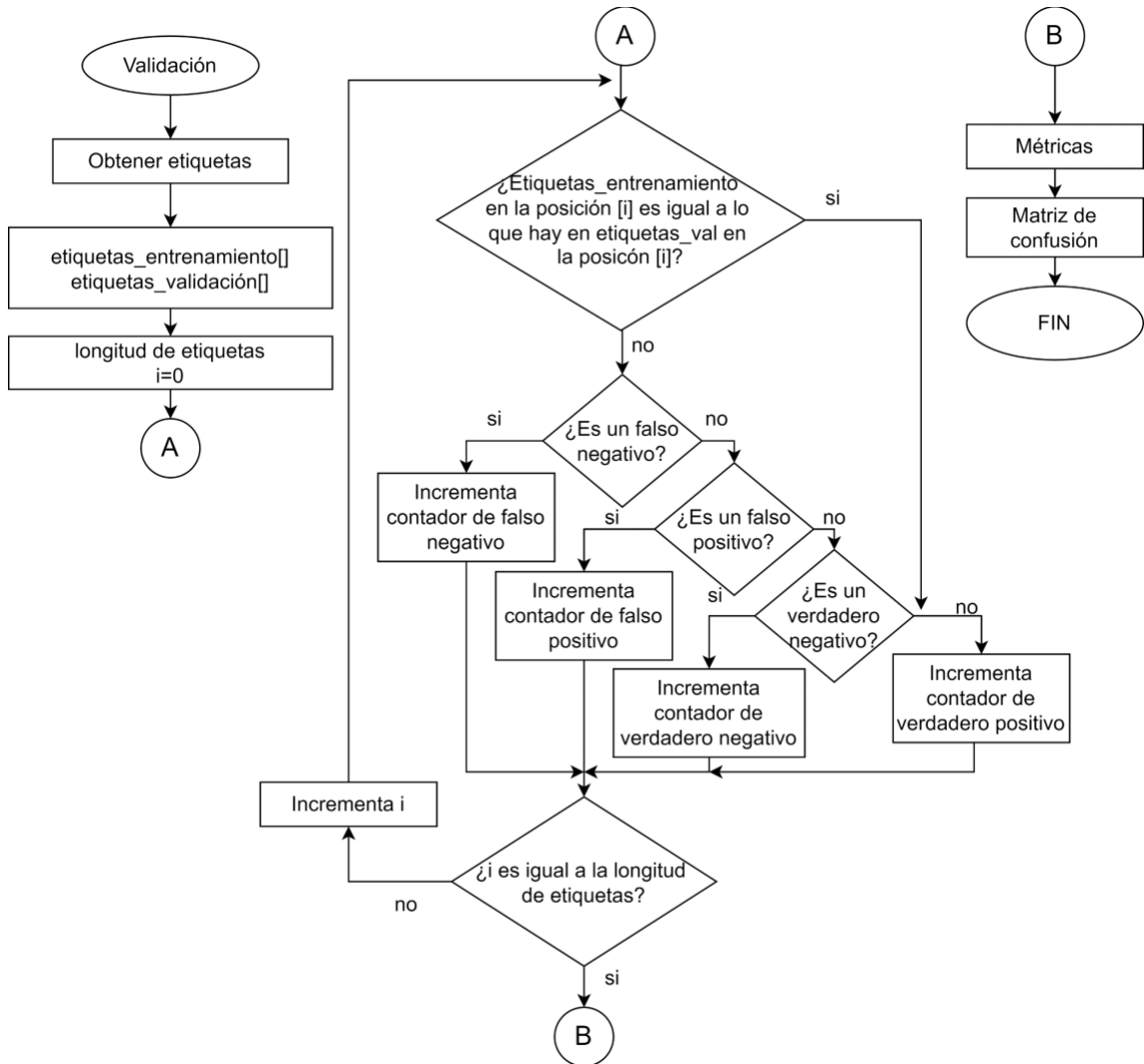


Figura 33. Proceso para matriz de confusión



Finalmente, el desarrollo de la red neuronal realizada en Python, al igual que las imágenes que se utilizaron para el entrenamiento y validación se encuentran en el repositorio de GitHub del siguiente enlace <https://github.com/MayraMateo0/Clasificacion-de-piezas-met-licas-CNN-PCA>.

#### 5.4.8 Desarrollo de interfaz para el manejo del sistema

El sistema clasifica las piezas metálicas regulares: círculo, rectángulo, cuadrado y rondana e irregulares: clavo y tuerca.

Para que el usuario pueda utilizar los archivos de la red neuronal convolucional con PCA realizada en *Google Colab* de manera local, se crean dos archivos, uno corresponde a pesos y el otro a la CNN, ambos con una extensión “.h5”.

Se cargan los archivos de pesos y de la red neuronal con extensión “.h5” mediante Python de manera local para hacer uso de la clasificación con la red neuronal con PCA. Para que el usuario pueda utilizar el sistema, se realiza una interfaz con dos funciones principales, una para clasificar las piezas metálicas en tiempo real y otra función para clasificar imágenes que se encuentren almacenadas en carpeta. Los formularios que se utilizan para la interfaz se muestran en el diagrama entidad relación de la Figura 34:

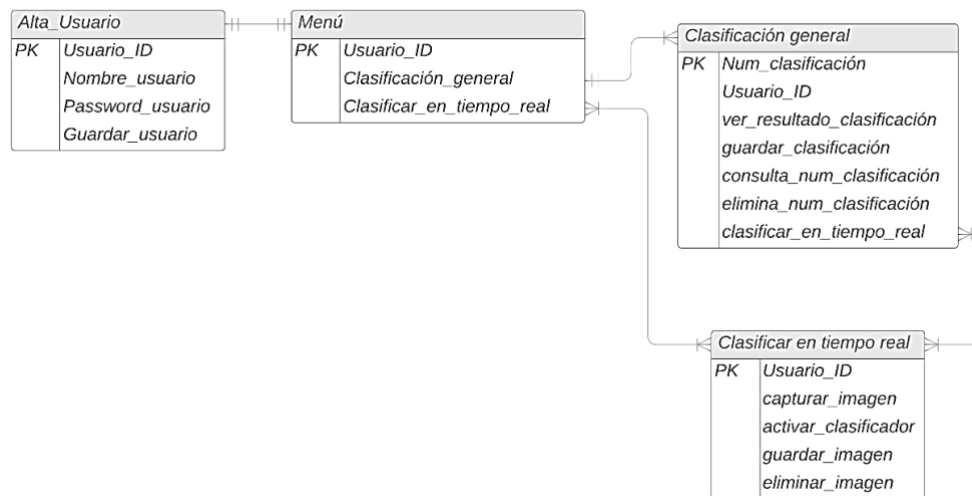


Figura 34. Diagrama entidad relación de la interfaz del sistema.

De acuerdo con el diseño de los formularios, se presentan los diagramas de maquetación Figura 35-38:

La Figura 35 corresponde a la ventana de inicio. Se muestra el esquema de cómo están colocados los logos, banner, botones, la imagen y el título de inicio.

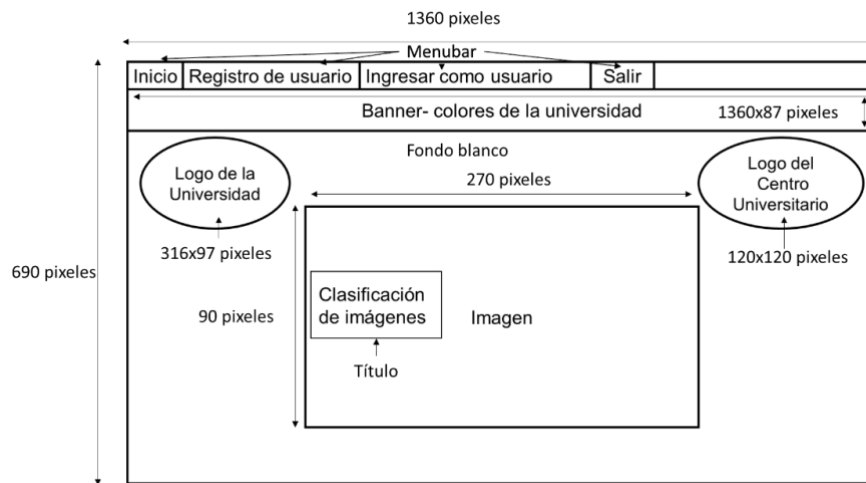


Figura 35. Diagrama de maquetación de inicio

Para acceder al sistema, el usuario debe registrarse para verificar quienes tienen acceso al sistema, por lo tanto, en la Figura 36 se muestra el esquema para el registro del usuario. Se colocaron dos cajas con la función "entry" de Python para introducir el contenido. Una vez colocado el contenido, solo letras para el nombre y solo números para la contraseña, se guarda el registro con el botón de *Guardar Registro*.

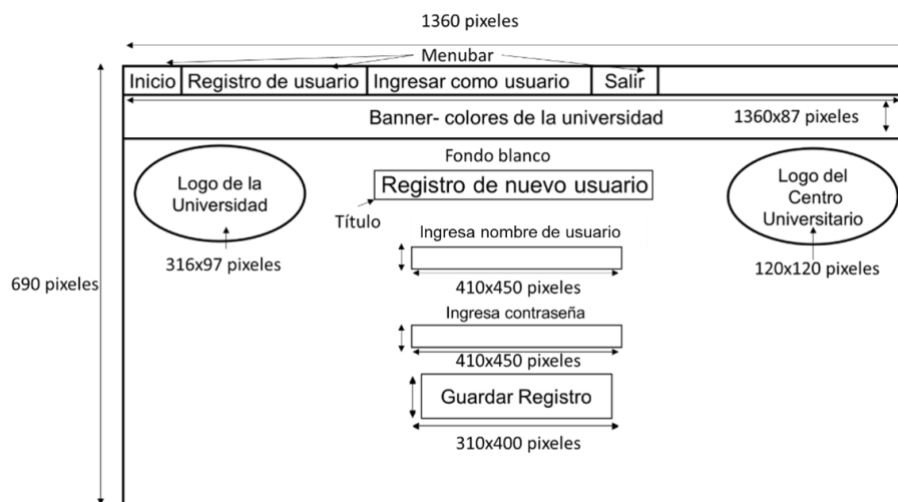


Figura 36. Diagrama de maquetación de Registro de nuevo usuario

La Figura 37 corresponde al menú para ingresar como usuario al sistema. Se colocaron dos cajas con la función “entry” de Python para introducir el contenido, para después presionar el botón de “Ingresar”.

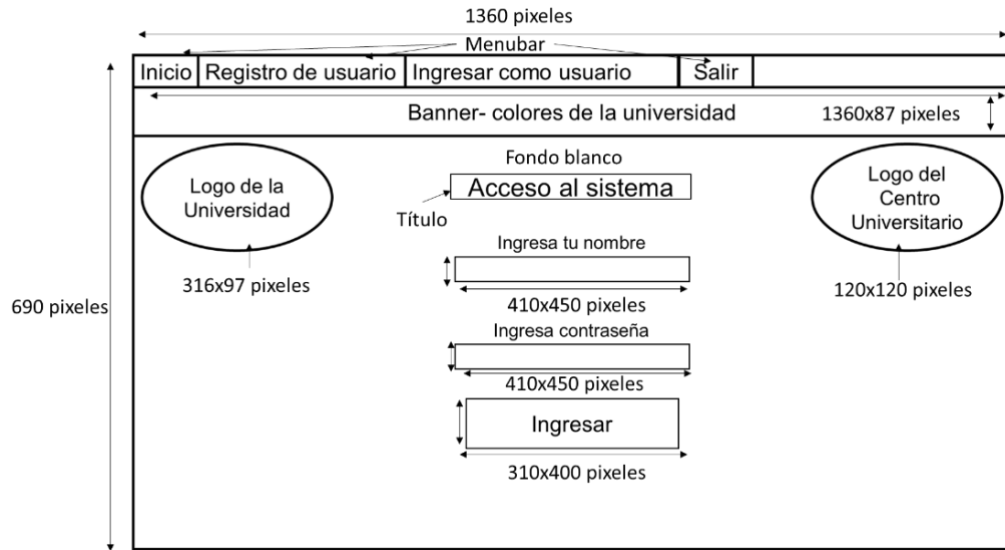


Figura 37. Diagrama de maquetación de Acceso al sistema.

El esquema que hace referencia a las funciones del sistema se muestra en la Figura 38, las cuales son dos botones, clasificar en tiempo real y clasificar total de imágenes contenidas en carpeta.

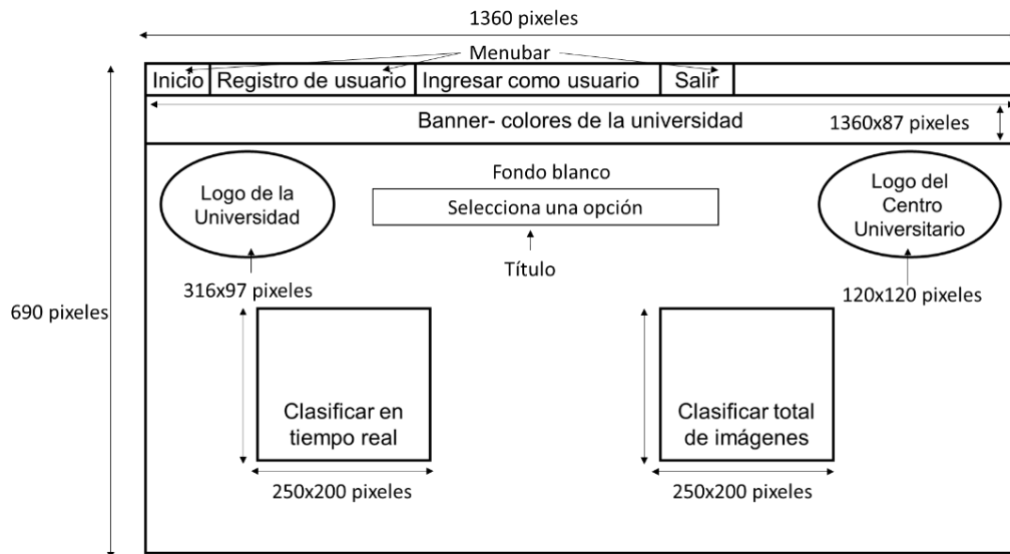


Figura 38. Diagrama de maquetación de funciones para el sistema.

El diagrama de la posición de los elementos del resultado de clasificación se muestra en la Figura 39.

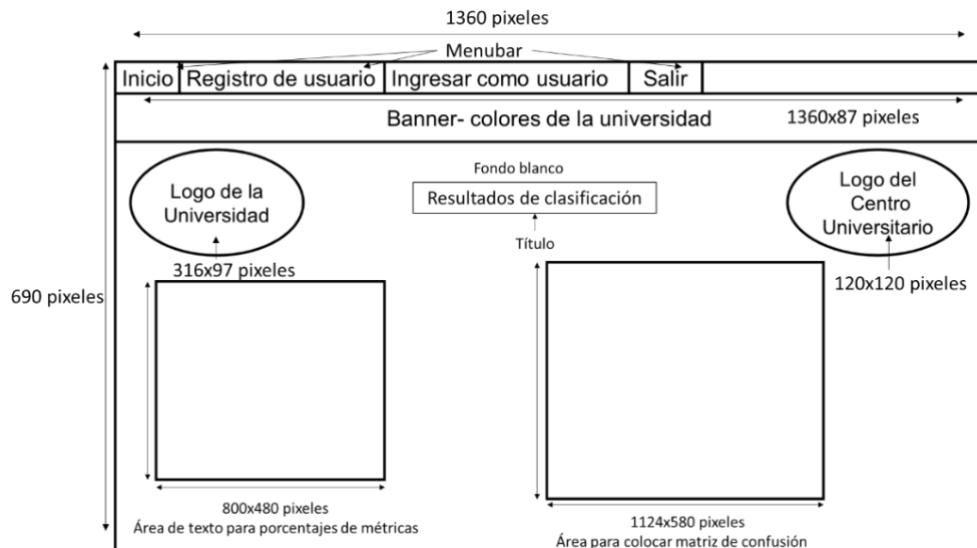


Figura 39. Diagrama de maquetación de los resultados clasificación con matriz de confusión.

Los resultados de clasificación de cada una de las imágenes contenidas en carpeta se colocaron de acuerdo con el diagrama mostrado en Figura 40. El área para mostrar imágenes sirve para desplegar todas las imágenes que se encuentren en carpeta, junto con la etiqueta proporcionada por el sistema de clasificación.

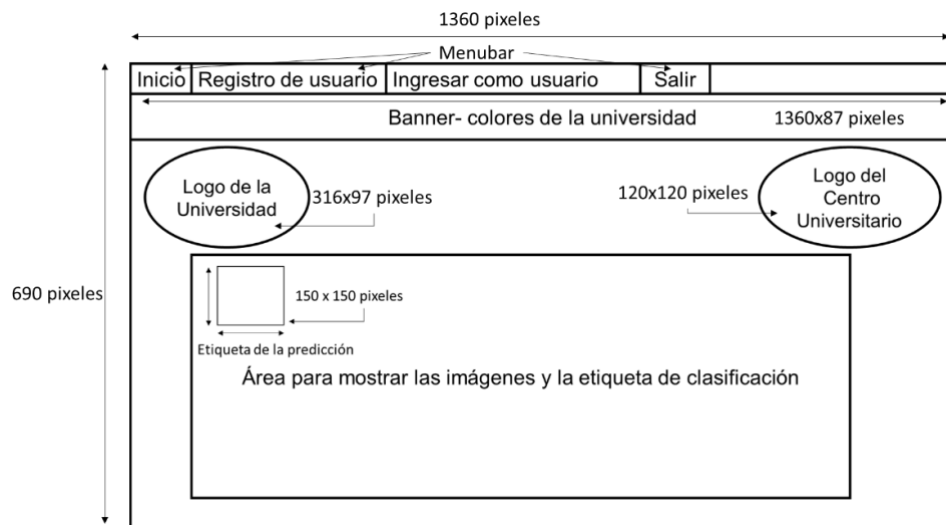


Figura 40. Diagrama de los resultados de clasificación de imágenes contenidas en carpeta.

Para representar el comportamiento del sistema, se presenta la *Figura 41*, que implica la representación del escenario de la interacción entre el usuario y el sistema, además de los mensajes que se generan al momento de dar clic en algún botón.

Se tienen tres formularios con las funciones que generan eventos, los cuales son registro de usuario, ingresar como usuario y acceso a usuario. El primer paso que realiza el usuario es la realización del registro a través de ingresar un usuario y contraseña, de los cuales el nombre no debe contener números y la contraseña no debe tener letras, de lo contrario se le envía el mensaje al usuario de que se genera un error. Cuando el usuario ingrese los datos de manera correcta entonces se despliega un mensaje en pantalla con la información que se almacenó, posteriormente, con esos datos ingresa al siguiente formulario que es el de ingreso como usuario, donde va a colocar el usuario y contraseña, si no es correcto, se despliega el mensaje de que no existe el usuario, en caso de acceder con los datos, se abre otro formulario que tiene las funciones del sistema, sin embargo, se muestra un mensaje para que el usuario verifique la cámara este conectada para que el sistema funcione de manera adecuada.

El formulario contiene la función de clasificar las piezas metálicas en tiempo real, donde a las imágenes se les aplica un pretratamiento mediante PCA para luego etiquetarlas conforme se coloquen debajo de la lente de la cámara, el nombre de la figura aparecerá en pantalla, después de haber realizado la clasificación con la Red Neuronal Convolutiva.

Para la opción de clasificar las imágenes que estén almacenadas en carpeta, las imágenes reciben un pretratamiento con PCA y posteriormente se muestra una imagen con todas las imágenes disponibles en la carpeta con su respectiva etiqueta de clasificación, además, se tiene la opción de generar datos estadísticos, los cuales se generan a partir de la creación de la matriz de confusión que tiene origen de la Red Neuronal Convolutiva, donde, al presionar el botón se despliega un mensaje y finalmente como resultado se muestra un nuevo formulario con la matriz de confusión y los datos de las métricas obtenidas en porcentajes.

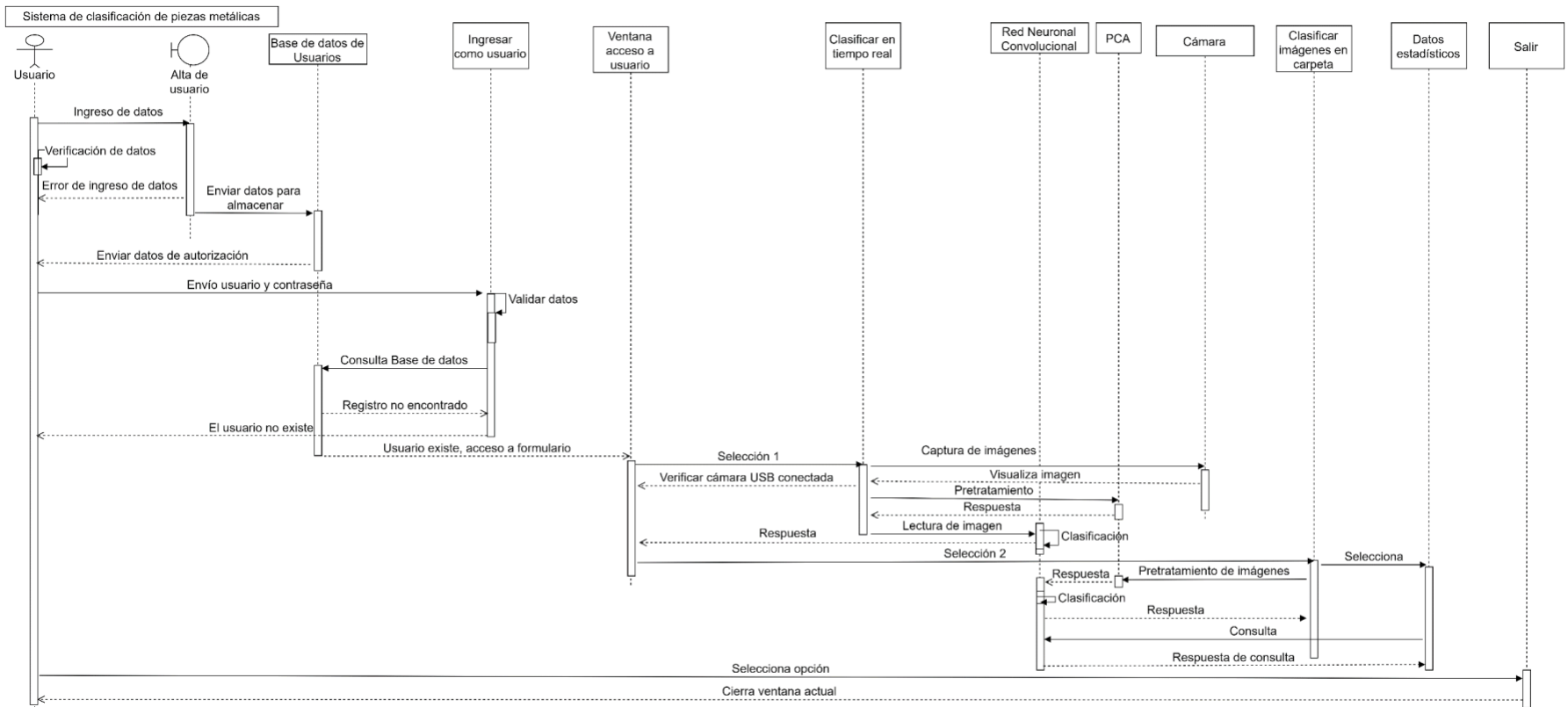
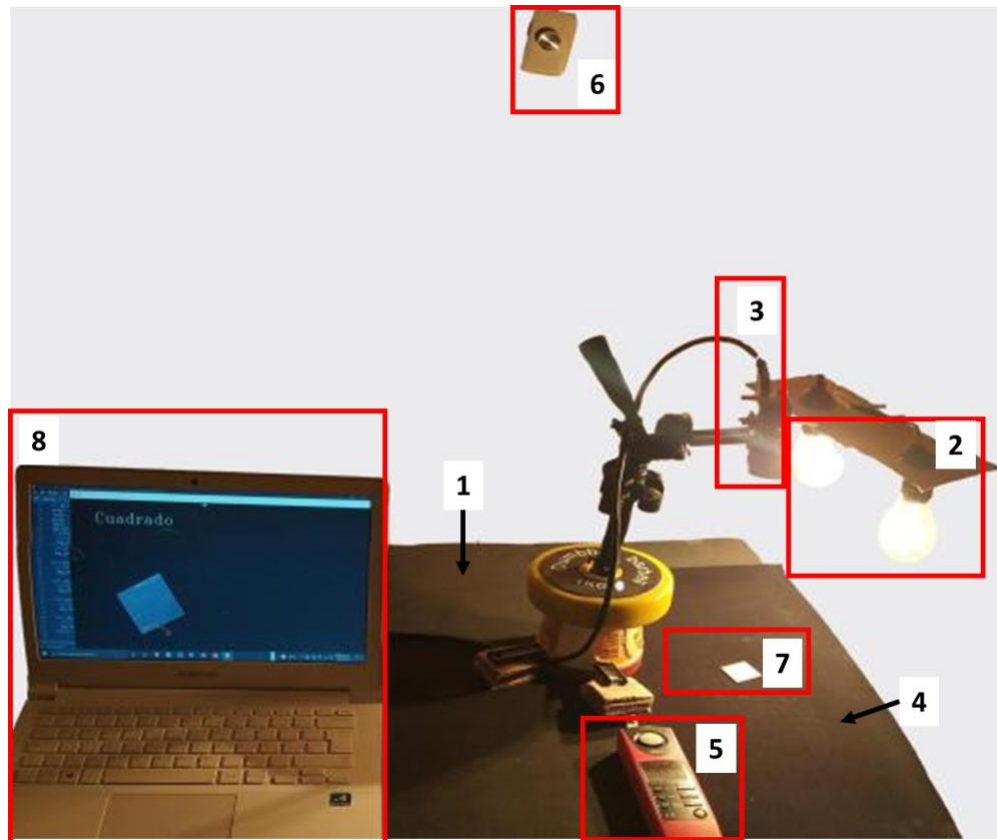


Figura 41. Diagrama de secuencias del sistema de clasificación de piezas metálicas.

## 6 RESULTADOS Y DISCUSIÓN

Para realizar la experimentación se adecuó el entorno de trabajo para colocar los componentes que se especifica en la Figura 27 de la sección de método. El resultado de la adecuación se presenta en la Figura 42:



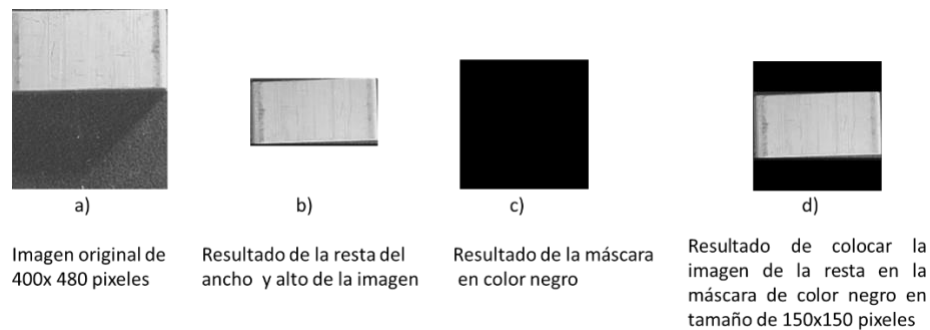
*Figura 42. Diseño del entorno de trabajo real. Los elementos de la imagen son: 1.-Mesa de trabajo, 2.- Lámparas de 100W, 3.- Cámara, 4.- Fondo negro, 5.- Luxómetro UT380 Series, 6.- Dimmer, 7.- Pieza de metal, 8.- Computadora donde se aloja el sistema y se conecta la cámara USB.*

Posteriormente, mediante el entorno de trabajo se realizó la captura de las imágenes de las piezas metálicas para después normalizarlas y que tengan el mismo tamaño de 150 x 150 píxeles, tal como se indica en la sección de normalización de imágenes.

### 6.1 Normalización de imágenes

Al obtener las imágenes con la cámara de 1.3 MP, se obtienen dimensiones de 300 x 200 píxeles y con la cámara de 8 MP se obtienen imágenes capturadas con 300 x 200 píxeles

y 400x480, por lo tanto, se normaliza el tamaño de la imagen a 150 x 150 píxeles, además, es conveniente reducir la carga de procesamiento de la red neuronal al analizar imágenes, y realizar el proceso para redimensionar las imágenes que comienza con la verificación de que las dimensiones alto y ancho sean distintas, puesto que en ocasiones existen dimensiones del mismo tamaño ya sea en ancho, alto o ambas debido a la forma de la figura, tal es el caso de la figura del cuadrado, círculo y rondana. El proceso de recorte se indica en el diagrama de la Figura 28, con el cual, se procesan imágenes como en el caso de la Figura 43, a. Luego se obtiene el recorte de la imagen de interés (Figura 43, b). Como la imagen resultante no tiene el tamaño de 150 x 150 píxeles, se crea una máscara de color negro y se pega la imagen encima (Figura 43, c). El resultado del proceso se indica en la Figura 43, d.

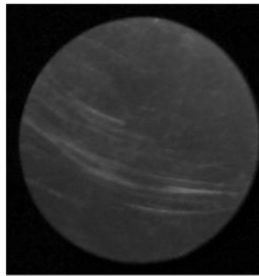


*Figura 43. Proceso de redimensionamiento.*

Debido a las diferentes exposiciones de niveles de iluminación algunas de las imágenes obtenidas se muestran opacas, esto puede representar un problema ya que los contornos pueden ser confundidos con el fondo. Para mejorar el brillo de manera automática y lograr que se definan los contornos de las figuras, se utiliza el proceso de balance de iluminación, expuesto en el método de la Figura 29. Un ejemplo de las piezas que presentó opacidad fue el círculo (Figura 44 a), la cual está redimensionada a 150 x 150 píxeles, el resultado al aplicar el balance de iluminación se presenta en la Figura 44 b).

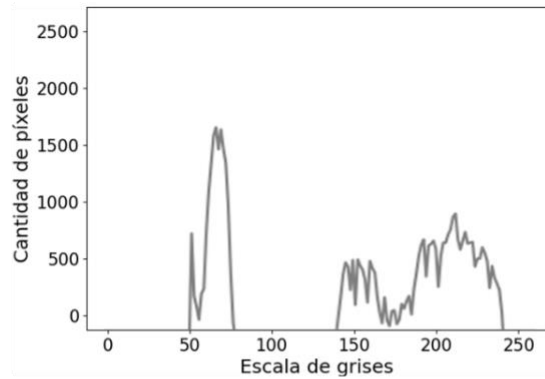


Imagen sin balance de iluminación



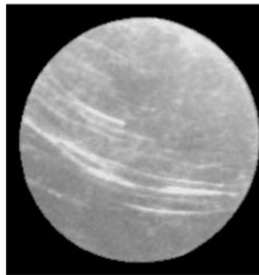
a)

Histograma sin balance de iluminación



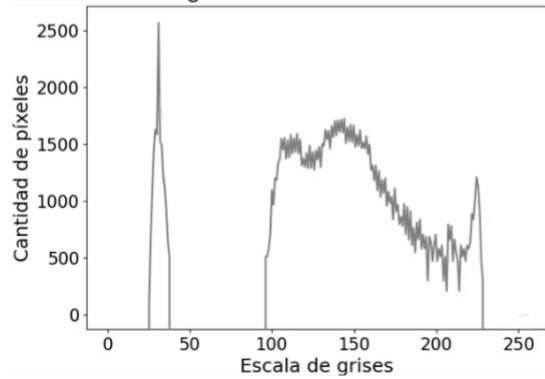
b)

Imagen con balance de iluminación



c)

Histograma con balance de iluminación



d)

*Figura 44. Resultados del balance de iluminación con la imagen del círculo a 1500 lx*

El histograma de Figura 44 a), se visualiza en la Figura 44 b), la distribución de los píxeles cuando se acercan al valor de 0 indica ausencia de luz, mientras que para 255 son intensidades más luminosas. Se observan dos regiones, una que va de 25 a 40 en escala de grises con frecuencia de 1600 píxeles como máximo, y otra con un intervalo más amplio (140 a 245 en escala de grises) con cantidades variantes de píxeles que no superan los 800.

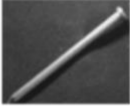
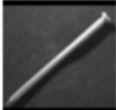
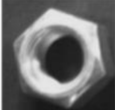
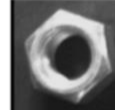

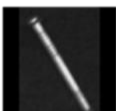
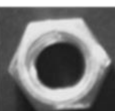
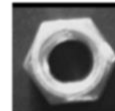

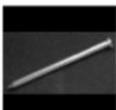
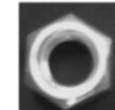
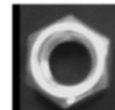
Por otra parte, en el histograma de la Figura 44 d), la distribución de los píxeles se agrupa en dos conjuntos, un primer conjunto comienza en 25 hasta 40 en escala de grises, puesto que representa valores oscuros, con una amplitud máxima de 2600 píxeles, mientras que el segundo conjunto comienza en 95 hasta 225 en escala de grises, representando las zonas con mayor iluminación, lo que quiere decir que la imagen resultante Figura 44 d) presenta un color de fondo negro más definido al igual que en los píxeles que aportan mayor intensidad luminosa que refleja la superficie del objeto y que en comparación de la figura

del inciso a), las tonalidades son más oscuras teniendo una frecuencia de pixeles menor a 800 para el caso del segundo conjunto.

La segunda etapa de la normalización consiste en calcular el promedio de los valores de iluminación de los pixeles de cada una de las imágenes para obtener el balance de la iluminación, en este caso el valor de iluminación promedio obtenido mediante pruebas fue de 100. Entonces se evalúan todas las imágenes y aquellas que estén por debajo del promedio se les aplica el algoritmo. En la Figura 45-44 se aprecian ejemplos de los resultados obtenidos del balance de iluminación en las imágenes de las piezas metálicas expuestas a diferentes niveles de iluminación. Uno de los resultados más notorios se presenta en la figura círculo de 1000 lx (indicado con un rectángulo rojo en Figura 45-44), porque se muestra más oscura debido a la posición en la que se colocó la pieza para capturar la fotografía; en la imagen resultante (indicada como balance de iluminación) se identifica el contorno de la pieza, característica que ayuda a definir de qué pieza metálica se trata. Las piezas que ocasionan falsos positivos o negativos fueron el rectángulo y el círculo. En el caso del rectángulo se debe a que el color de la pieza original es de un tono dorado y al momento de exponerla a luz cálida se pierden los bordes de la pieza en el tono de fondo, como es en el caso de 2000 lx. Respecto al círculo, la tonalidad del fondo se asemeja al de la pieza y hace que el borde del círculo se integre como parte de la pieza. En el caso de la rondana, cuadrado, tuerca y clavo, se identifican los bordes de manera adecuada.

Rango de iluminación(lx)	Figuras regulares							
	Original	Balance de iluminación	Original	Balance de iluminación	Original	Balance de iluminación	Original	Balance de iluminación
1000								
1500								
2000								

Figura 45. Muestra de balance de iluminación con luz cálida en imágenes de 1.3 MP en cada nivel de iluminación (1).

Rango de iluminación(lx)	Figuras irregulares			
	Original	Balance de iluminación	Original	Balance de iluminación
1000				
1500				
2000				

*Figura 46. Muestra de balance de iluminación con luz cálida en imágenes de 1.3 MP en cada nivel de iluminación (2).*

Para los siguientes casos de exposición de luz fría y cálida, se presentan las imágenes más representativas por provocar mayor cantidad de falsos positivos y negativos. En el caso de las imágenes de 1.3 MP con iluminación fría (Figura 47) son, el rectángulo porque el color de fondo tiene una tonalidad parecida a la de la pieza original en los tres niveles de iluminación, lo que hace difícil la distinción entre los contornos de la pieza original con el fondo. Otro ejemplo de las piezas es la tuerca porque tiene una profundidad en sus dimensiones haciéndola una pieza en 3D, esto ocasiona que no se ilumine de manera uniforme y que solo resalte un área de la pieza haciendo que se pierda con la tonalidad de fondo y generando una mayor iluminación como es en el caso de 2000 lx. Además, al aplicar luz fría, las piezas toman una tonalidad blanca por lo tanto reflejan la luz que incide, tomando una intensidad luminosa superior a la que debería tener la pieza metálica en la superficie, esto ocasiona que no se identifiquen de manera correcta.

Nivel de iluminación (lx)	Piezas metálicas (iluminación fría)			
	Original	Balance de iluminación	Original	Balance de iluminación
1000				
1500				
2000				

Figura 47. Balance de iluminación en imágenes de 1.3 MP con iluminación fría.

Para las imágenes de 8 MP con iluminación fría y cálida, se ven con intensidad luminosa superior en ambos casos, incluso se nota la diferencia de la tonalidad del fondo y de la pieza. En el caso de las imágenes con iluminación cálida, las imágenes se ven suavizadas con 1000 y 2000 lx, lo que provocaría que no se clasifiquen correctamente y existan más caso de verdaderos y falsos positivos. Las imágenes con iluminación fría reflejan en mayor cantidad la luz respecto a la superficie del objeto, dificultando la distinción entre el fondo y el borde la imagen. Se puede concluir que la luz cálida proporción una tonalidad opaca a diferencia de la iluminación fría, donde el fondo y la pieza se muestran con mayor intensidad luminosa.

Nivel de iluminación (lx)	Piezas metálicas (iluminación cálida 8MP)				Nivel de iluminación (lx)	Piezas metálicas (iluminación fría)			
	Original	Balance de iluminación	Original	Balance de iluminación		Original	Balance de iluminación	Original	Balance de iluminación
1000					1000				
1500					1500				
2000					2000				

a)

b)

Figura 48. Balance de iluminación en imágenes de 8 MP a) iluminación cálida, b) iluminación fría.

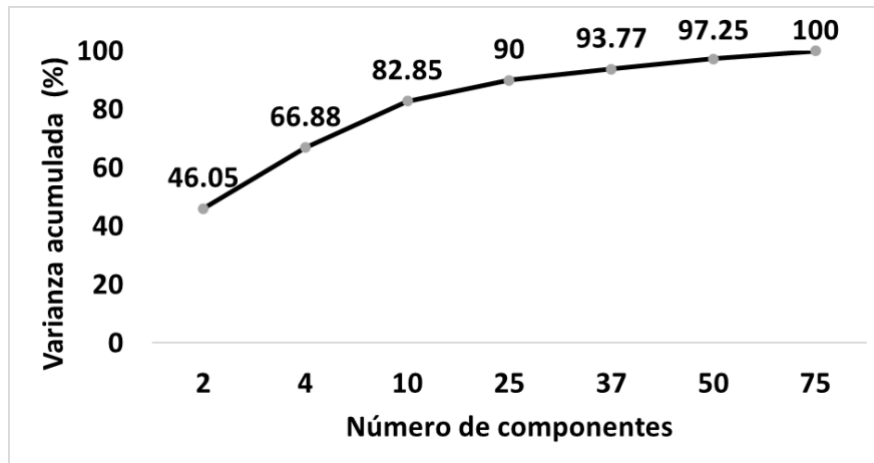
Por lo tanto, las imágenes a utilizar son las capturadas con la cámara de 1.3 MP con iluminación cálida porque aporta un mejor contraste entre la pieza y el fondo, lo que permite la identificación del borde.

Después de la normalización de imágenes se divide el dataset para entrenamiento (60 %) y validación (40 %). Los datos de entrada para la red neuronal hasta este punto son de una matriz de tamaño 648 x 432 que corresponde a la cantidad de imágenes de entrenamiento y validación. El tamaño de la matriz es igual a la cantidad de componentes y se procede a realizar el pretratamiento de imágenes con PCA para reducir la cantidad de componentes.

## 6.2 Pretratamiento de imágenes con PCA

El algoritmo de PCA se aplica a todas las imágenes y se obtiene el total de componentes que es de 480, sin embargo, es conveniente analizar los valores menores a 50 componentes (Figura 49). Por cada uno de los componentes se obtiene una varianza acumulada mediante la ecuación (4), sin embargo, en Python, se utiliza la sentencia “*cov\_matrix = np.matmul (standardized\_data.T, standardized\_data)*”, de la librería de *scikit-learn*, la cual indica la cantidad de información para graficar una imagen. Para saber cuál es la cantidad adecuada para reconstruir las imágenes, se utilizaron 2, 4, 10, 25, 37 y 50 componentes para evaluar la cantidad de exactitud que genera en la red neuronal convolucional mediante las pruebas de validación.

La varianza acumulada para los componentes seleccionados se visualiza en la Figura 49 , el valor mínimo se tiene con 2 componentes con 46.05 % de varianza y como máximo se tienen en 50 componentes con 97.25 % de varianza. El valor del 100 % de varianza acumulada se obtiene a partir del valor de 75 componentes, sin embargo, el valor más cercano fue de 50 componentes, la cual es una cantidad menor para considerar, ya que uno de los objetivos es evaluar la menor cantidad de componentes. Para reducir carga computacional en la RNC se requiere graficar con la menor cantidad de componentes sin diferir significativamente la imagen resultante de la original.



*Figura 49. Varianza acumulada de los componentes PCA*

Al realizar las pruebas con la red neuronal convolucional, se ejecutaron 10 iteraciones con 50 épocas. Para analizar la distribución de los datos respecto al valor promedio de los datos y determinar si existe algún tipo de sesgado, ya sea a la izquierda, derecha o distribución normal, se utiliza un gráfico de cajas y bigotes.

Para todos los casos presentados, un sesgado a la izquierda indica que existe una mayor dispersión de los valores hacia valores bajos (0 %) y que se obtiene una exactitud mínima, por lo tanto, este tipo de dispersión se analiza de acuerdo con el valor mínimo que alcance. Existe un sesgado a la izquierda cuando la distribución de los valores se aproxima al valor más alto (100 %). Se identifica el color morado que corresponde al cuartil 3 (Q3) que representa el 75 % de la distribución de los valores, en caso de “x” representa la media, el color verde indica el cuartil 1 (Q1) que es el 25 % de la distribución de los valores.

Por cada una de las iteraciones se obtuvieron los resultados de acuerdo con las métricas de exactitud, sensibilidad, precisión y F1. Dentro de los parámetros de la red neuronal convolucional (nivel de iluminación, optimizador, número de componentes PCA), se sustituye el valor de número de componentes PCA, primero por 2 componentes, los resultados se graficaron en la Figura 50.

Para la métrica de exactitud se obtuvo el valor promedio de las 10 iteraciones que corresponde al 76.49 % como valor de tendencia central. Esto indica que las 10 iteraciones se acercan al valor promedio de exactitud, haciendo más probable que las iteraciones siguientes obtengan un valor de exactitud aproximado.

Para la métrica de sensibilidad, el producto del cálculo del promedio de 10 iteraciones corresponde al 76.86 %, indicando que es el porcentaje de la capacidad de la red neuronal convolucional para detectar imágenes de manera correcta.

El resultado del promedio de las 10 iteraciones respecto a la métrica de precisión es del 75.50 %, las cuales muestran un sesgado a la izquierda, porque los valores que van desde la media, se despliegan hacia valores cercanos al 60 % de exactitud, existiendo mayor dispersión de datos, además se observa un comportamiento inestable porque el valor promedio (línea azul) varía en las iteraciones presentadas, identificando también la dispersión de los valores con longitudes de bigotes de diferente tamaño, lo cual indica que es más probable que se obtengan resultados de clasificación por debajo del valor promedio de la precisión.

En el caso de la métrica F1, el porcentaje promedio de las 10 iteraciones corresponde a 76.18 %. Evalúa la relación entre precisión y sensibilidad e indica el valor promedio ponderado de las dos métricas. Con el uso de dos componentes no existen datos atípicos, porque no existen puntos fuera de los bigotes, ni tampoco en los rangos límites que van del 40 % (valor mínimo) hasta el 100 % (valor máximo).

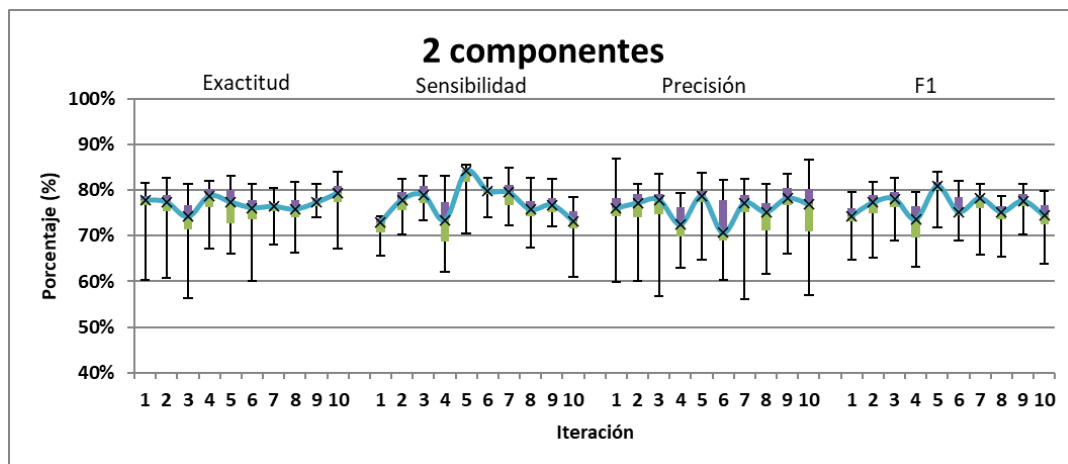


Figura 50. Iteraciones con 2 componentes.

Posteriormente, los resultados con 4 componentes se ilustran en Figura 51. Para la métrica de exactitud, los datos tienen un sesgado a la izquierda, es decir, que, los resultados tienen mayor distribución hacia valores bajos a partir del valor promedio que es del 76.71 %, implicando que existen mayor cantidad de datos en comparación de valores que superen

el promedio de exactitud. Respecto a la métrica de sensibilidad, se obtuvo el promedio de las 10 iteraciones, resultando el 78.42 %; sin embargo, a partir de la iteración 6, el promedio fue disminuyendo, hasta la iteración 10 con un valor del 73 % de sensibilidad, con un sesgado a la izquierda, así que, es más probable que las siguientes iteraciones se encuentren en un rango del 85 % hasta un valor mínimo del 63 %.

El porcentaje promedio para la precisión es del 75.50 %. Al analizar el comportamiento de las iteraciones para el valor del cuartil 2 se encuentra en el 78 %, en el caso de la iteración tres, mientras que en la siguiente iteración es del 77 %, después vuelve a subir y por consiguiente, el comportamiento se repite para las demás iteraciones siguientes. Lo anterior indica una inestabilidad en los porcentajes resultantes, por lo tanto, tendrá un promedio del 75.50 %.

Mientras que para la métrica F1 se obtuvo el 76.96 % del promedio de las 10 iteraciones. También existe una inestabilidad, lo cual indica que mientras se tenga un porcentaje de sensibilidad alto y un porcentaje de precisión bajo, la red neuronal convolucional detecta la clase, pero es poco confiable (porcentaje de la precisión) en las clasificaciones realizadas.

De acuerdo con la capacidad de la red neuronal convolucional, solo clasificará el total de imágenes en un 76.71 %, pero la similitud de las imágenes a las que proceden de las clases será del 75.50 %, entonces al tener un porcentaje bajo indica que dejará pasar mayor cantidad de falsos negativos y positivos en un 25 %, lo que hace poco confiable el uso de la red neuronal convolucional.



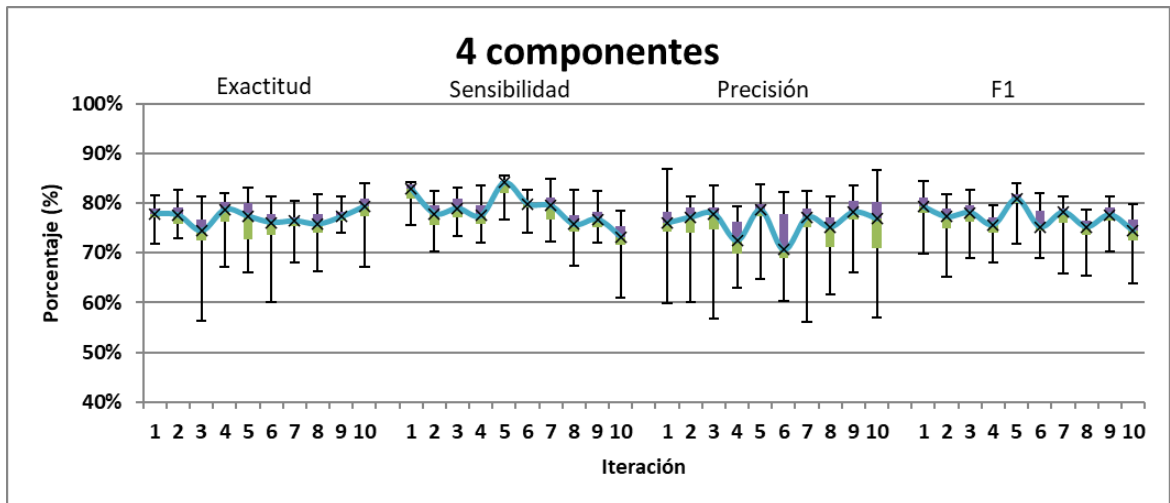


Figura 51. Entrenamiento con 4 componentes.

En la Figura 52 (uso de 10 componentes), el promedio las 10 iteraciones para la exactitud es del 81.09 %, sin embargo, la dispersión de los datos disminuyó, entonándose como límite el valor mínimo del 70 %, por lo tanto, los valores se encuentran más concentrados entre el cuartil 1, 2 y 3; también muestra estabilidad para las iteraciones 9 y 10. Esto indica que es más probable obtener el 81.09 % de exactitud por cada diez iteraciones. En el caso de la sensibilidad, el porcentaje es del 80.53 %, sin embargo, los datos se encuentran dispersos, donde, las iteraciones 4,5,6 y 9, muestran un sesgado a la izquierda, así que es poco probable que la red neuronal identifique las clases de manera correcta. En el caso de la precisión, se obtuvo el promedio del 79.65 %, donde las diez iteraciones tienen un sesgado a la izquierda, lo cual hace que la clasificación realizada con la red neuronal no sea confiable o que deje pasar falsos positivos y falsos negativos. Para la métrica F1 corresponde el resultado del promedio de 10 iteraciones con el 80.09 %. Es importante señalar que los datos tienen un sesgado hacia la izquierda, es decir, hay mayor cantidad de datos que son menores al promedio, lo que ocasiona que la red neuronal identifique las clases, pero con poca confianza, equivalente al porcentaje de la precisión.

Por ejemplo, si se tuvieran 120 imágenes, solo el 81.09 % será clasificado de manera acertada, que corresponde a 97 imágenes, pero con un 79.65 % de similitud a las imágenes originales, lo que origina mayor probabilidad de obtener falsos positivos y negativos con el porcentaje restante del 18.91 % para clasificar.

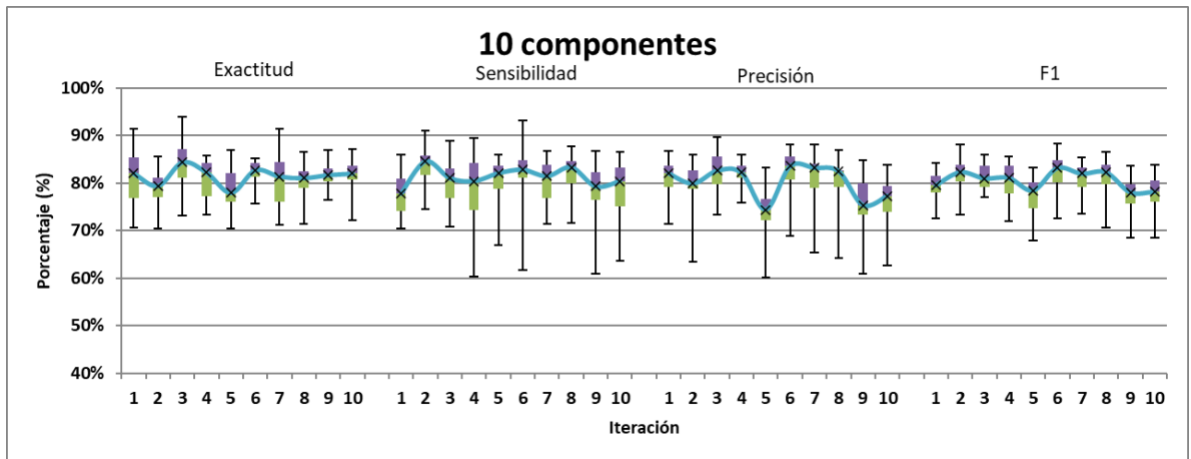


Figura 52. Entrenamiento con 10 componentes.

En las pruebas realizadas con 25 componentes (Figura 53) disminuyó la dispersión de los datos, obteniendo los siguientes promedios de las 10 iteraciones; exactitud 95.7 %, sensibilidad 95.22 %, precisión 95.80 %, F1 95.50 %, con valores altos de precisión y de sensibilidad, indicando que la red neuronal identifica correctamente las clases y que además es confiable en las predicciones que realiza. Lo anterior indica que el uso de 25 componentes es adecuado de acuerdo con los porcentajes obtenidos durante las 10 iteraciones, además, la estabilidad de los resultados en exactitud indica que se tendrán promedios cercanos al 95.7 %, lo que quiere decir que, por 120 imágenes a clasificar, solo el 5 % será clasificado de manera incorrecta, que equivale a seis imágenes aproximadamente, consideradas como falsos positivos o falsos negativos.

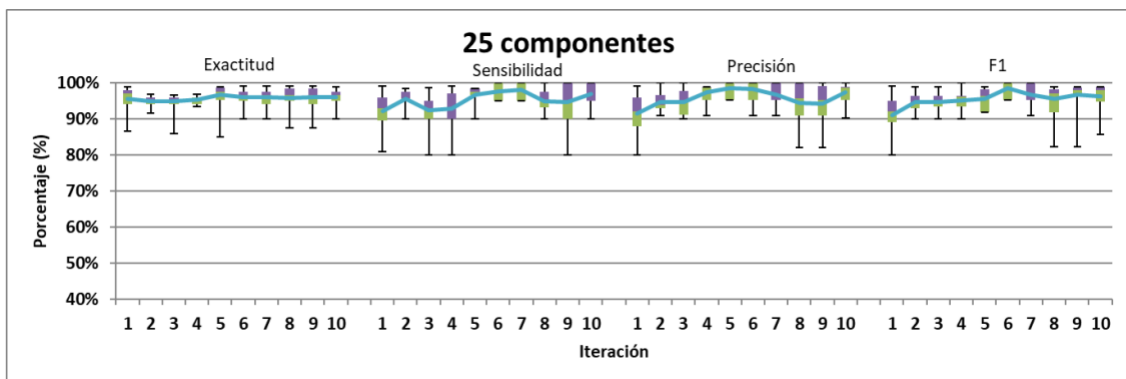


Figura 53. Entrenamiento con 25 componentes.

El uso de 37 componentes se ilustra en la Figura 54, donde el promedio de las 10 iteraciones corresponde a: exactitud 90.22 %, sensibilidad 88.58 %, precisión 85.45 % y

F1 87.02 %. A pesar de que el valor de la exactitud es alto, el promedio de sensibilidad, precisión y F1, bajan, incluso, en iteraciones como en el caso de la exactitud, se muestra un sesgado a la izquierda, con valores mínimos hasta el 85 %. En la sensibilidad existen valores más bajos que alcanzan el 73 % que indica mayor dispersión de datos. En la precisión, los datos son dispersos, lo que indica que no identifica de manera correcta las imágenes y que solo el 85.58 % de que las imágenes sean similares a las reales, pero solo acertará el 90.22 % de imágenes correctas. Para la métrica F1, los valores de sensibilidad y precisión indican que la red neuronal tiene la capacidad de detectar la clase correctamente y que tiene una confianza equivalente al 87.02 %.

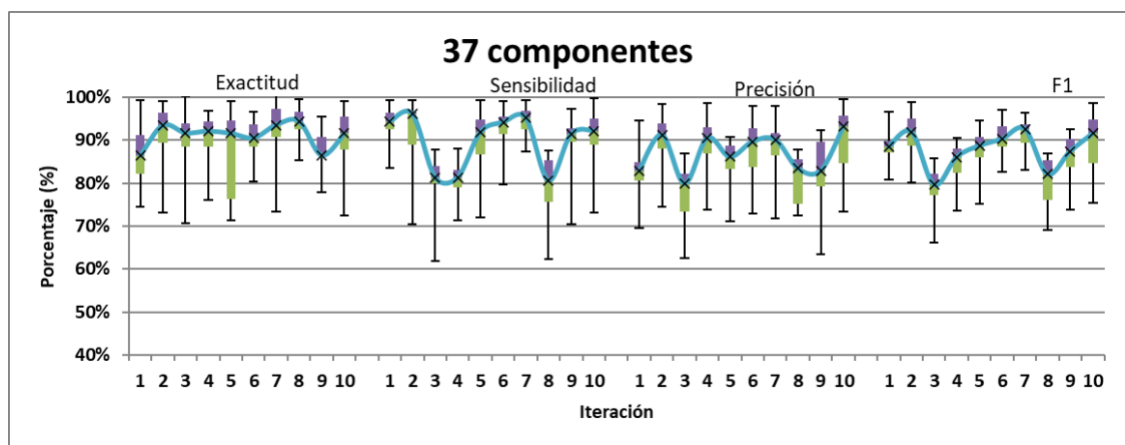


Figura 54. Entrenamiento con 37 componentes.

Los promedios que corresponden a la Figura 55 son: exactitud 83.20 %, sensibilidad 83.20 %, precisión 83.20 %, F1 83.20 %. Para las métricas de sensibilidad, precisión F1, los datos están estables, alrededor del 83.20 %, sin embargo, la métrica de exactitud tiene mayor inestabilidad en las iteraciones, mostrando un sesgado a la izquierda, así que es más probable que se obtengan valores respecto al valor promedio y que solo en 2 de cada 10 iteraciones se alcancen porcentajes del 90 %. Por ejemplo, en caso de la iteración 4, con la precisión se obtiene el 90 %, pero en sensibilidad se obtiene el 81 %, precisión 80 %, F1 80 %, lo que indica que, a mayor precisión, es menor la cantidad de clases a detectar y menor la confianza obtenida de la RNC y mientras se tengan valores promedio del 83 %, los valores de las métricas se van a conservar entre esos valores, lo cual indica que la red neuronal convolucional tenga mayor cantidad de verdaderos positivos y negativos.

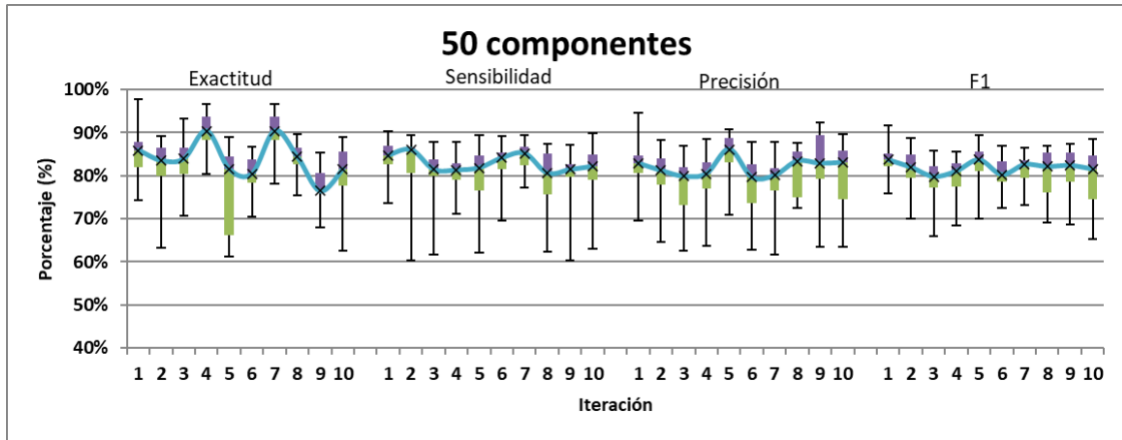


Figura 55. Entrenamiento con 50 componentes.

De los resultados obtenidos, para el uso de componentes menores a 25, se obtienen porcentajes por debajo del 80 % de exactitud, sin embargo, con el uso de 25 componentes generan mejores resultados de exactitud 95.7 %, sensibilidad 95.22 %, precisión 95.80 %, F1 95.50 %, descartando así el uso de 37 componentes, puesto que los datos están concentrados y dispersos lo que hace que el porcentaje sea bajo para todas las métricas.

En la Figura 56, se analiza por cada componente el valor de la exactitud, ya que es la métrica de interés en este trabajo, sin embargo, en el apartado de validación se analizan las métricas de precisión, F1 y sensibilidad. El eje x representa los valores de componentes evaluados y el eje y representa el porcentaje, que corresponde a la gráfica de exactitud que se obtuvo por cada uno de los componentes. El porcentaje mínimo en la gráfica es para el uso de 2 componentes con 76.49 % y el porcentaje más alto es de 95.75 % con 25 componentes. Por lo tanto, el uso de componentes 2, 4 y 10 no son adecuados porque se obtienen porcentajes menores al 90 %. Para componentes mayores a 25, por lo menos con 37 componentes se obtiene el 90.22 % de exactitud, sin embargo, durante las pruebas con 37 componentes, los resultados de los datos tuvieron una desviación estándar del 6.32 % (Tabla 11) a partir del promedio de los datos que es del 90.22 %. En el caso de 25 componentes, le corresponde un promedio del 95.75 %, con una desviación estándar del 2.4 %, lo que demuestra que los datos están menos dispersos y es más probable obtener promedio de clasificación cercanos al 95.75 % de exactitud.

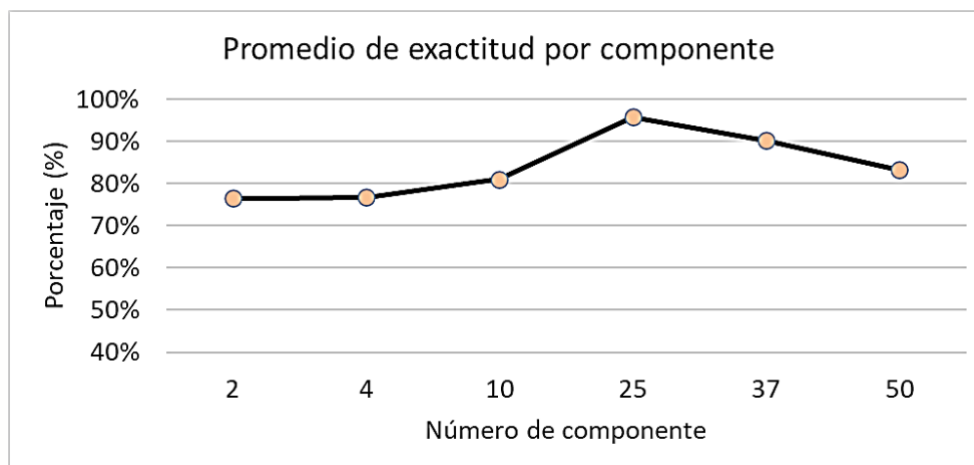


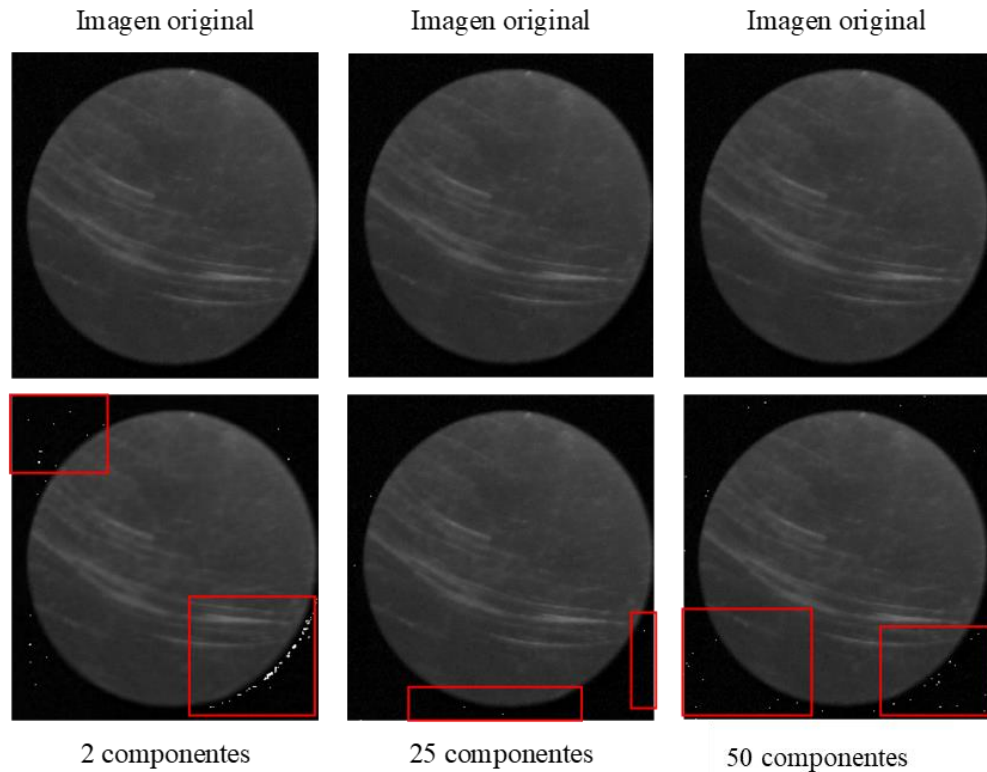
Figura 56. Promedio de las métricas de 2, 4, 10, 25, 37 y 50 componentes.

La desviación estándar por cada componente se muestra en la Tabla 11. El menor porcentaje de desviación estándar corresponde a 25 componentes con 2.40 %, lo cual indica que es más probable que los resultados de exactitud se concentren de acuerdo con el valor promedio obtenido, mientras que la desviación estándar con porcentaje más alto es para 37 y 50 componentes, puesto que la desviación estándar aumenta, causando que los resultados generados sean poco confiables por la dispersión de los datos.

Tabla 11. Desviación estándar de la exactitud obtenida por cada componente.

Componentes	2	4	10	25	37	50
Desviación estándar (%)	3.75	3.42	3.02	2.40	6.32	6.53

El efecto que tiene la aplicación de componentes PCA sobre las imágenes, se visualiza a través de la reconstrucción o proyección de imágenes. La Figura 57 muestra la aplicación de 2, 25 y 50 componentes en la imagen del círculo. En las tres imágenes con componentes se observa el ruido de sal y pimienta (recuadros rojos), en forma de puntos blancos donde es más notorio en la aplicación de 2 componentes, al igual que con 50 componentes, aunque esta última contiene menor cantidad de ruido en comparación de 2 componentes. Mientras que, con 25 componentes existe menor cantidad de ruido en comparación con 2 y 50 componentes.



*Figura 57. Proyección de componentes.*

La proyección o graficación de las imágenes sirvió para verificar que no tuvieran ruido como en el caso de 2 y 50 componentes en el fondo, por lo tanto, con 25 componentes se considera adecuado por la menor cantidad de ruido que contiene.

### **6.3 Resultados del diseño factorial 3<sup>4</sup>**

Para visualizar la distribución de los datos con relación a las variables: componentes (PCA), número de neuronas en la capa oculta y nivel de iluminación, se utilizó un gráfico de barras en 3D (Figura 58-59), con sus respectivos ejes, para representar una de las variables de estudio que es la exactitud. Las gráficas se agrupan de acuerdo con el tipo de cámara de 1.3 y 8 MP, por niveles de luxes y por el uso del optimizador SGD, puesto que se obtuvieron mejores resultados de exactitud.

En la Figura 58 a) se presenta la gráfica resultante para el caso de las imágenes de la cámara con 1.3 MP, empleando 1000 lx y optimizador SGD, se observa la exactitud con un promedio alto del 95.56 % y un promedio bajo del 70.83 % mientras que en la Figura 58 b) para imágenes de 8 MP el promedio alto de 71.94 % y bajo del 67.22 % de exactitud.

De lo anterior se deduce que, con imágenes expuestas a 1000 lx, el mejor resultado se obtiene con la cámara de 1.3 MP con 95.56 % de exactitud con las variables de 16 neuronas en capa oculta, y 50 componentes y optimizador SGD, sin embargo, se evalúa si los resultados mejoran con 1500 lx o 2000 lx.

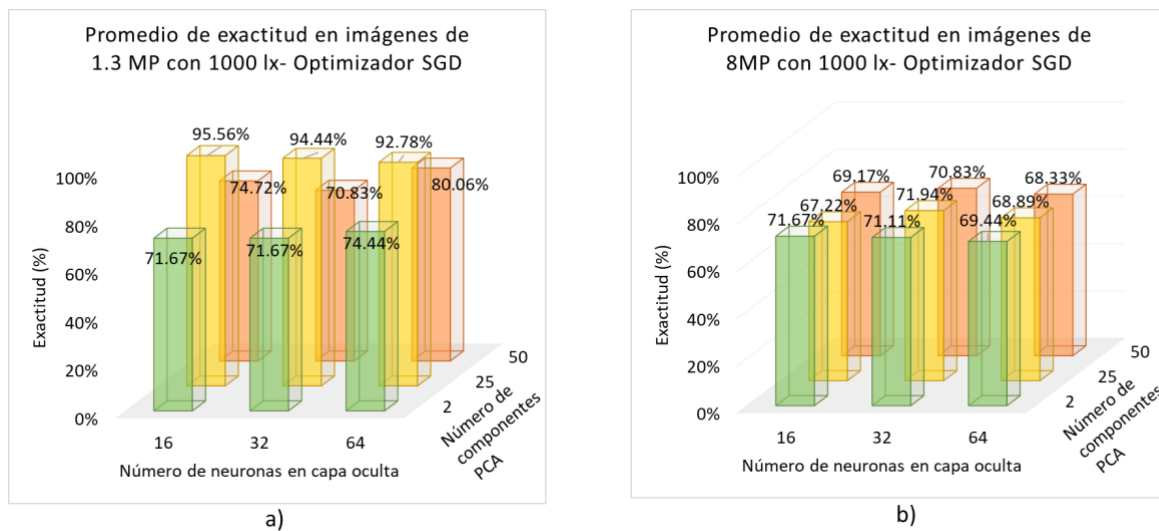
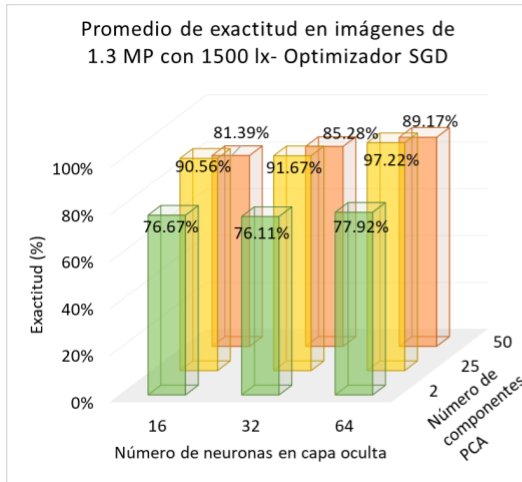


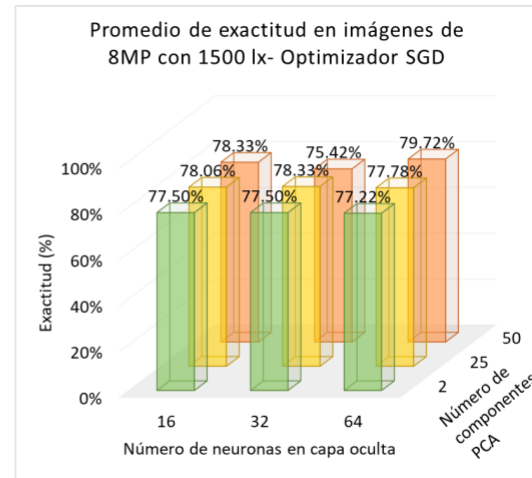
Figura 58. Gráficas del promedio de exactitud, con 1000 lx. a) imágenes de 1.3 MP. b) imágenes con 8 MP.

En la Figura 59 a) se presentan los resultados con imágenes de 1.3 MP a 1500 lx, la mejor combinación respecto al promedio fue cuando se emplean 64 neuronas en capa oculta, 25 componentes (PCA) además, se utilizó el optimizador SGD, obteniendo un promedio del 97.22 % de exactitud, mientras que la clasificación con el porcentaje más bajo con un promedio del 76.11 % de exactitud resulta de la combinación de 32 neuronas en capa oculta, optimizador SGD y 2 componentes (PCA).

Los resultados para las imágenes de 8 MP con 1500 lx se presentan en la Figura 59 b), la mejor combinación presenta un promedio del 79.72 % de exactitud, empleando 64 neuronas en capa oculta, optimizador SGD, 50 componentes (PCA). El resultado más bajo para la métrica de exactitud corresponde al 77.22 %, con 64 neuronas en capa oculta, optimizador SGD, 2 componentes (PCA).



a)



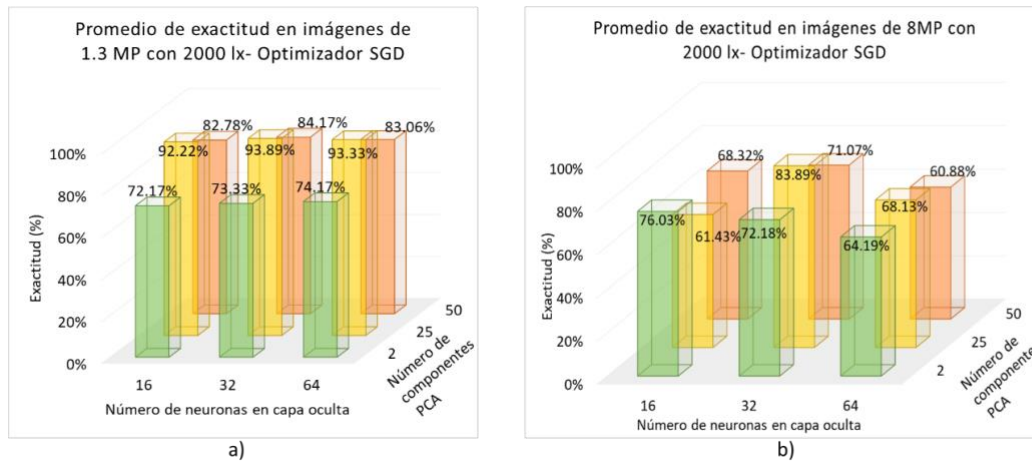
b)

Figura 59. Gráficas del promedio de exactitud, con 1500 lx. a) imágenes de 1.3 MP. b) imágenes con 8 MP.

Para el gráfico Figura 60, a) el valor más bajo de exactitud es del 72.17 % y el valor más alto corresponde al 93.89 %. Por lo tanto, los valores del promedio de exactitud que se obtuvo en Figura 59, a) fue del 97.22 % y respecto a los valores de Figura 60, a), con 93.89 % de exactitud, solo existe una diferencia del 3.33 %. La diferencia se presenta en la cantidad de imágenes que se clasifiquen de manera acertada, por ejemplo, de 120 imágenes, el 3.33 % corresponde a 4, de las cuales, serían correctamente clasificadas con el 97.22 %, entonces, se considera como mejor promedio para realizar las clasificaciones.

En el caso de las imágenes de 8 MP, se obtuvo un valor bajo del 63.36 % y un valor alto del 83.89 % de exactitud. En comparación del valor del promedio anterior de exactitud que fue del 75.42 % y del valor actual, la diferencia es del 8.47 %. Aunque el valor del porcentaje aumentó, la exactitud para clasificación es mejor con imágenes de 1.3 MP.





*Figura 60. Gráficas del promedio de exactitud, con 2000 lx. a) imágenes de 1.3 MP. b) imágenes con 8 MP.*

De lo anterior se deduce que el mejor resultado para la métrica exactitud fue con el uso de 1500 lx, 25 componentes PCA, 32 neuronas en capa oculta y el uso del optimizador SGD. El uso del optimizador SGD se determinó por las pruebas presentadas en Figura 22 y porque en el diseño factorial 3<sup>4</sup>, el porcentaje es de 97.22 % de exactitud. El resultado de una iteración es la matriz de confusión de la Figura 61 con la misma combinación, únicamente cambiando el optimizador por RMSProp y Adam. Para la Figura 61 a), se utiliza el optimizador Adam, la clasificación de las 20 imágenes correctas corresponde a las clases cuadrado, rondana y círculo, sin embargo, existen 10 imágenes correctamente clasificadas para clavo, 11 imágenes para rectángulo y 19 para tuerca, las demás imágenes se encuentran distribuidas en clases distintas. Para el optimizador RMSProp se muestra la Figura 61 b), donde existe mayor cantidad de clasificaciones correctas, como es el caso de 20 imágenes clasificadas por cada caso respecto a clavo, cuadrado, rondana, círculo y tuerca, únicamente en el caso de rectángulo se clasificaron correctamente 12 imágenes y 7 se identificaron como cuadrado, debido a que la figura se confunde con el fondo de la imagen. Con el uso del optimizador Adam se obtuvo una exactitud del 83.33 % y con el optimizador RMSProp con 92.5 % de exactitud, por lo tanto, se descartó el uso de los optimizadores y se utilizó SGD para realizar el entrenamiento con la red neuronal convolucional.

De los dos resultados, el mejor fue con el optimizador RMSProp, sin embargo, el resultado de clasificación es mejor con SGD, al obtenerse el 97.22 % de exactitud.

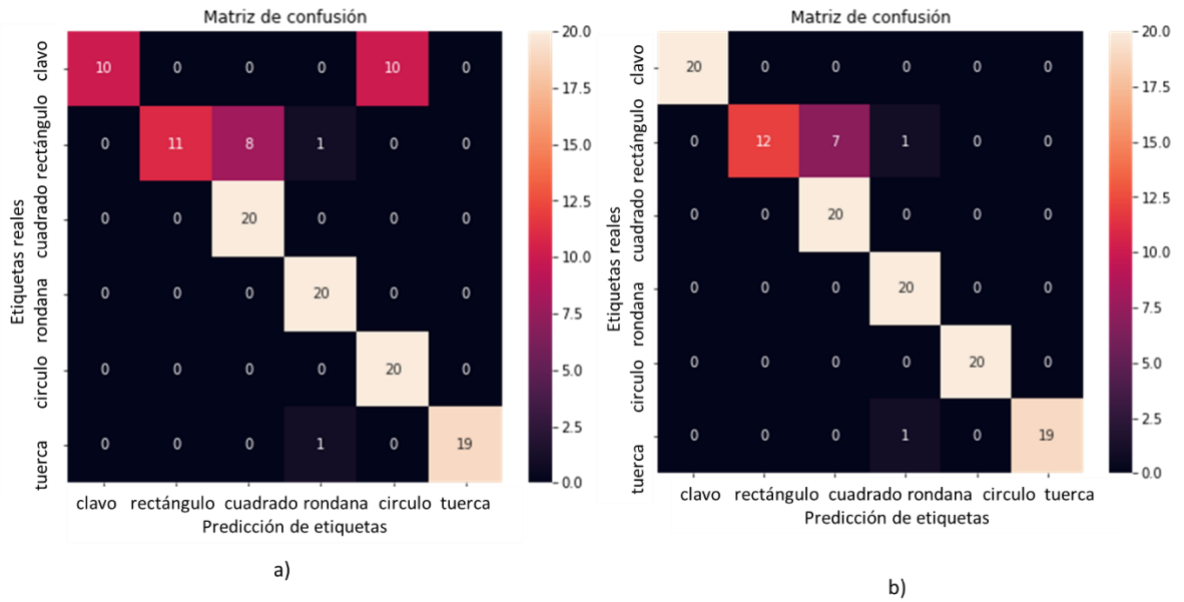


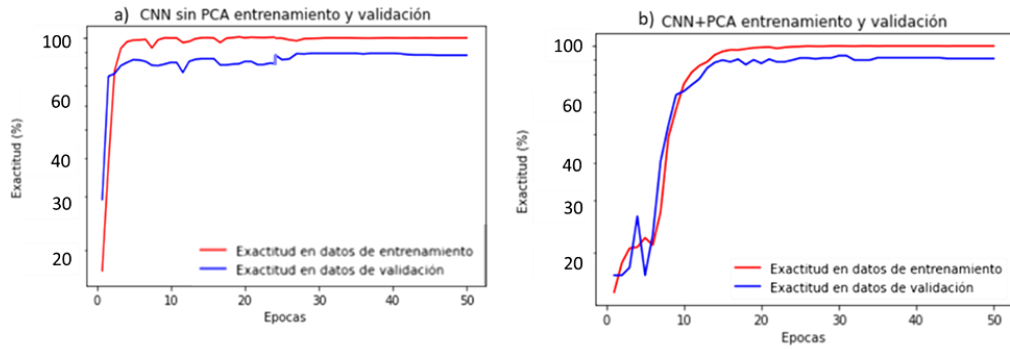
Figura 61. Resultados con optimizador a) Adam y b) RMSProp.

## 6.4 Evaluación de los resultados de entrenamiento con y sin PCA

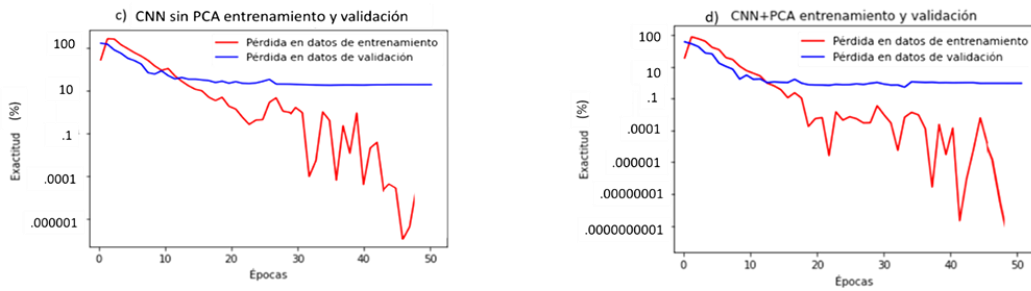
Para realizar el entrenamiento y determinar la pérdida de la RNC con y sin PCA, se ejecutó una iteración con 50 épocas, debido a que si se realizan otras iteraciones el comportamiento de convergencia es similar (Figura 62), por lo tanto, se muestran dos iteraciones para corroborar el comportamiento. Sin embargo, solo se analiza el entrenamiento y pérdida de la Figura 63.

Con el uso de la función “*fit()*” de *Keras*, se almacenan los valores que corresponden al historial de entrenamiento de la red neuronal convolucional, midiendo la métrica de exactitud durante la iteración del número de épocas totales, también mide la cantidad de pérdida que existe durante las iteraciones, ambos parámetros se aplican en los datos de entrenamiento y validación.

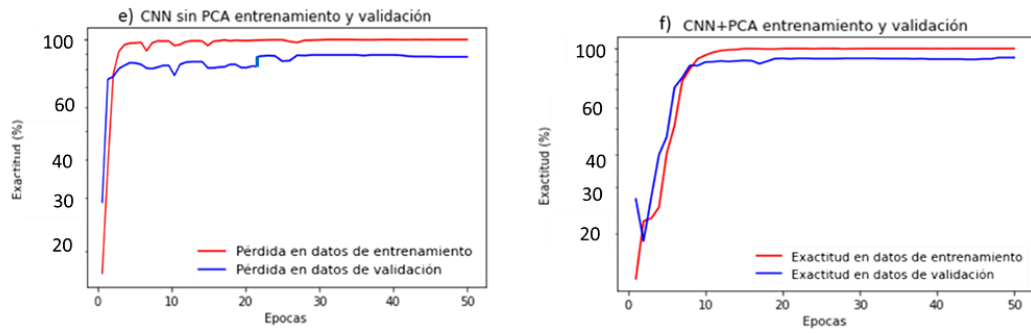
Gráficas de exactitud en entrenamiento y validación



Gráficas de pérdida en entrenamiento y validación



Gráficas de exactitud en entrenamiento y validación



Gráficas de pérdida en el entrenamiento y validación

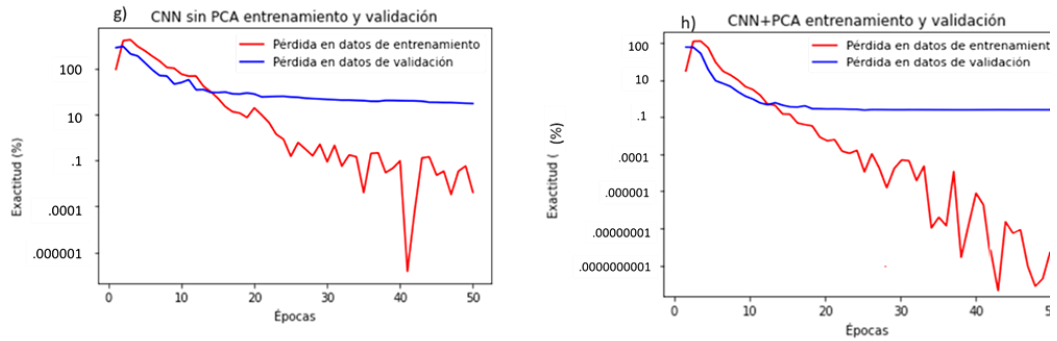


Figura 62. Entrenamiento y pérdida, con y sin PCA

El resultado del comportamiento para la métrica de exactitud de la RNC sin y con PCA se muestran en las gráficas de la Figura 63 a) y c), donde se presenta la exactitud en los datos de entrenamiento y validación; b) y d) corresponde a la pérdida en el entrenamiento de validación.

En la Figura 63 a), se aprecia que a partir de la época 20 la exactitud en el entrenamiento converge en el 100 %, mientras que los datos de validación a partir de la época 40 se van aproximando hasta llegar al 85 %. Sin embargo, se recomienda realizar más iteraciones para evaluar el parámetro de convergencia, puesto que la tendencia de la exactitud en el caso de la validación va aumentando. Con el comportamiento de las dos gráficas se deduce que la RNC no está sobre ajustada, ya que por cada iteración en la validación la tendencia va aumentando.

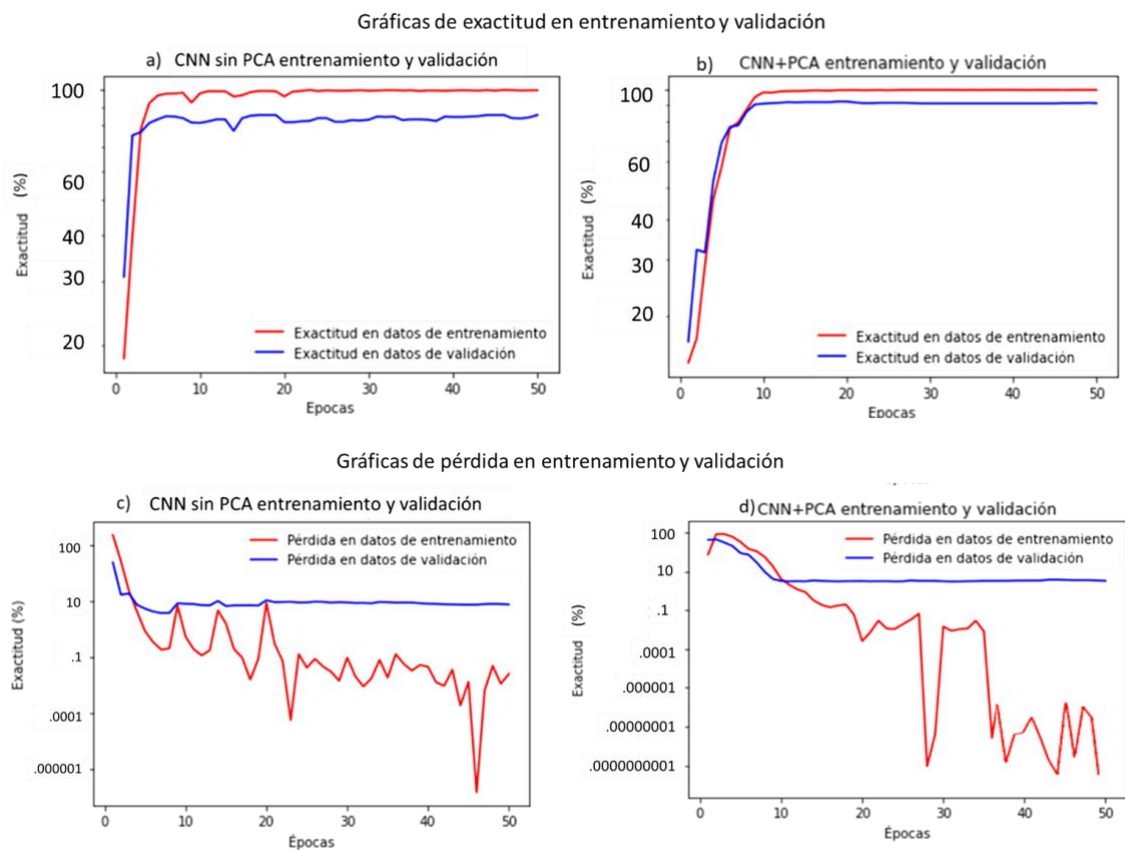


Figura 63. Resultados de entrenamiento y pérdida sin PCA: a) y c). con PCA: b) y d)

Las gráficas de la Figura 63 c y d, representan la pérdida que indica el porcentaje de imágenes que no clasifica de manera correcta. En la Figura 63 c), la pérdida de los datos

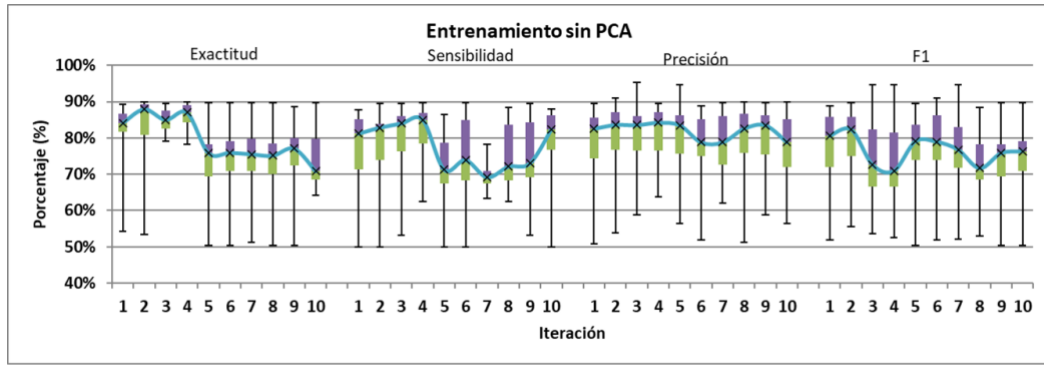
de entrenamiento es menor a comparación de los datos de validación, con valores cercanos al .1 % de exactitud. El porcentaje indica la desviación estándar de clasificaciones, donde la mayor cantidad de pérdida se aproxima al .0001 %. Se recomienda agregar más iteraciones para evaluar si la pérdida en la validación decrece o aumenta, puesto que en un principio va decreciendo, pero no se estabilizan los valores como es en el caso de los datos de entrenamiento.

Para el entrenamiento de la red neuronal convolucional con PCA, se observa que en la Figura 63 b), los porcentajes de los datos de entrenamiento respecto a la exactitud converge en el 100 %, mientras que para los datos de validación se acercan a una exactitud del 95 %. La tendencia de los datos indica que no existe un sobre entrenamiento porque empieza a converger en las últimas épocas, donde existe una estabilidad en el entrenamiento y validación, obteniendo la mejor solución para cada uno de los casos.

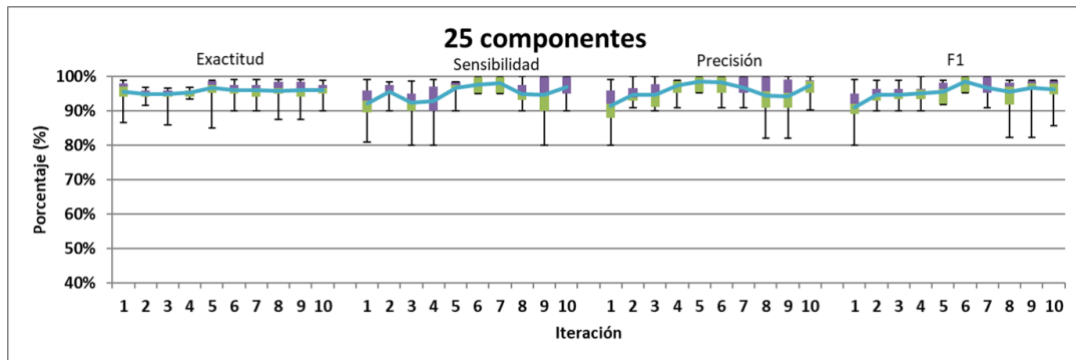
La pérdida del algoritmo RNC+PCA se muestra en la gráfica de la Figura 63 d). En comparación de los resultados de la gráfica sin PCA (Figura 63 c), el error en los datos de entrenamiento disminuye a valores de .0000000001 % para la última época. Es importante señalar que la diferencia visual que se observa entre ambos datos se debe a la escala logarítmica que se está utilizando.

Para evaluar la tendencia de los datos en los dos casos (con y sin PCA), se consideran 4 métricas: exactitud, sensibilidad, precisión y F1. Los valores de las métricas se graficaron en un diagrama de cajas y bigotes para detectar el sesgo y la tendencia en la que se distribuyen los datos.

En la Figura 64 a), la tendencia de los datos de exactitud, sensibilidad, precisión y F1, muestran un sesgado a la izquierda, lo que indica que la clasificación de los datos se encuentra dispersos y con porcentajes por debajo del valor promedio. La mayor cantidad de datos con porcentajes bajos reduce la posibilidad de tener un mejor promedio para cada una de las métricas, lo que indica que el porcentaje de clasificación puede ser menor al promedio, porque en la mayoría de las épocas existe una dispersión en datos hasta el 50 % de exactitud como mínimo.



a)



b)

Figura 64. Resultados de entrenamiento. Sin PCA a) y con PCA b)

La Figura 64 b), tienen un sesgado a la izquierda, pero a diferencia de los componentes 2, 4, 10, 37 y 50, existe mayor concentración de datos dentro de los cuartiles, lo que indica que existen menor cantidad de datos dispersos y son más cercanos al valor promedio. En la exactitud se observa un comportamiento estable respecto al valor promedio. La precisión indica que identifica las clases de manera correcta con un porcentaje del 91.4 %, con un promedio bajo de falsos positivos y negativos indicado por la sensibilidad del 92.22 %. Además, del total de imágenes va a detectar el 94.87 % de manera correcta.

Los promedios generales de las gráficas a) y b) de la Figura 64, se muestran en la Tabla 12, los resultados con el uso del algoritmo de PCA son mayores para cada una de las métricas, lo que indica que el algoritmo clasifica de mejor manera al emplear PCA. La métrica F1 indica que por lo menos el 98.57 % de imágenes son clasificados de manera correcta careciendo de falsos positivos y negativos.

Entonces con el entrenamiento que se obtiene con la red neuronal convolucional, solo permite identificar el 86.30 % de las imágenes totales, las cuales tendrán una similitud del 81.72 %, lo que significa que tendrá 14 % de falsos positivos y negativos, con una confianza del 78.75 %.

Los promedios máximos y mínimos de las métricas de precisión, exactitud, sensibilidad y F1, de ambos algoritmos se indican en la Tabla 12. Los mejores resultados corresponden al algoritmo de RNC con PCA, se obtiene un promedio del 96.83 % de exactitud, con una precisión del 98.57 %, con el 1.3 % de falsos positivos y negativos en clasificación, con una confianza del 98.57 %. Los valores anteriores se consideran porcentajes altos porque están más cerca del valor ideal que es el 100 %, por lo tanto, se determina que es mejor el rendimiento de la red neuronal con PCA. El promedio permite conocer la cantidad representativa de cada una de las métricas, así mismo, se considera como valor de medida central, lo cual es relevante para conocer el valor más alto de los resultados con y sin PCA, entonces se obtienen los valores máximos y mínimos para determinar los mejores y peores resultados obtenidos, llegando a la conclusión que con PCA se obtiene el mejor promedio de cada una de las métricas. En esta sección no se calcula la desviación estándar de los datos porque anteriormente en la Figura 64 se analizó en las gráficas de cajas y bigotes, los resultados con y sin PCA, determinando el tipo de sesgado, lo cual indica la dispersión de los datos, donde, los resultados sin PCA presentan mayor dispersión, sin embargo, para la sección de validación se incluye la desviación de estándar del conjunto de pruebas destinadas para este apartado, siendo relevante porque son los resultados finales de la experimentación.

*Tabla 12. Resultado de promedios en las métricas durante el entrenamiento*

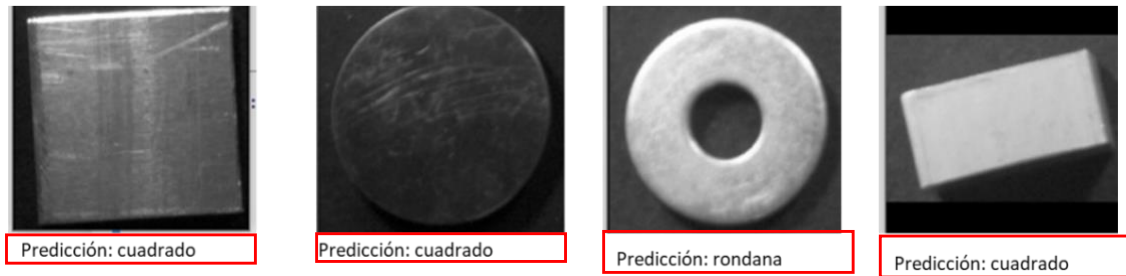
Promedio (%)	Exactitud (%)		Sensibilidad (%)		Precisión (%)		F1 (%)	
	Sin PCA	Con PCA	Sin PCA	Con PCA	Sin PCA	Con PCA	Sin PCA	Con PCA
<b>Max (%)</b>	86.30	96.83	82.37	98.1	81.72	98.57	78.75	98.57
<b>Min (%)</b>	74.20	94.87	70.21	92.22	77.34	91.40	73.23	91.04

Los porcentajes antes mencionados con PCA, son utilizados para clasificar los datos de validación a través de una matriz de confusión. Los resultados obtenidos se encuentran en la sección de validación.

## 6.5 Validación

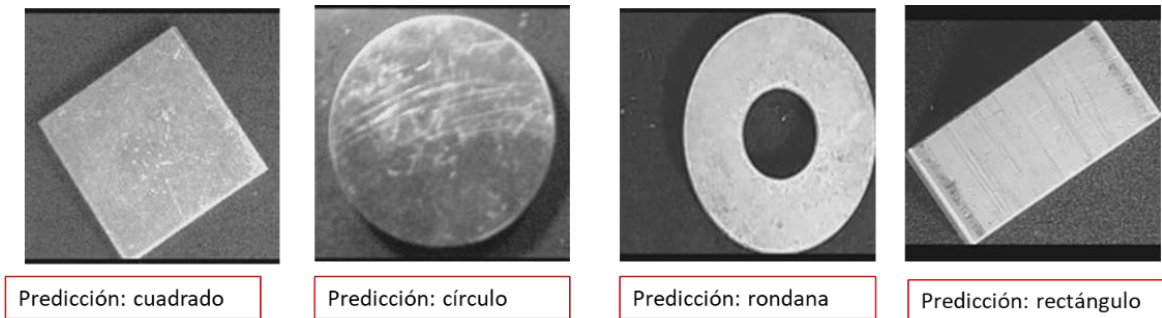
En la etapa de validación de imágenes del *dataset* se utiliza el 40 % de imágenes. El conjunto de imágenes está normalizado y redimensionado a escala de 150 x 150 píxeles. Las etiquetas que se utilizan son las que se obtuvieron en la sección de entrenamiento, las cuales son; 0: rectángulo, 1: cuadrado, 2: círculo, 3: rondana, 4: clavo, 5: tuerca.

Los resultados de clasificación se visualizan en la Figura 65, en donde se tiene un ejemplo de las imágenes de validación para clasificar y en la parte inferior se encuentra el resultado de clasificación por la Red Neuronal Convolutiva (referido con un rectángulo rojo). La imagen del círculo y del rectángulo no fueron correctamente clasificadas, debido a que las imágenes no cuentan con el balance de iluminación.



*Figura 65. Predicciones realizadas.*

Aplicando el balance de iluminación en las imágenes de las piezas metálicas (incluyendo el algoritmo de PCA), el resultado de clasificación para las piezas mejora, clasificando de manera correcta cada una de las imágenes, como se muestra en la Figura 66.



*Figura 66. Prueba de predicciones con balance de iluminación.*

Sin embargo, como se cuenta con 120 imágenes para validación, se utilizó la matriz de confusión para clasificar los resultados a través de la comparación de etiquetas reales con



las etiquetas de predicción, además, su uso permitió la evaluación de métricas como la exactitud.

Como parte del resultado de clasificación, con imágenes de 1.3 MP, se muestra una matriz de confusión (con y sin PCA), donde por cada una de las clases se utilizan 20 imágenes para validar. La diagonal principal que se muestra en tonos rosas indica el mayor porcentaje, es decir, indica el acierto de las etiquetas correspondientes con cada intersección de las clases que tienen en predicción y valor real.

En la matriz de confusión de la Figura 67, se observa que de 20 imágenes de la clase clavo, diez se clasifican correctamente, mientras que las diez restantes se clasifican como círculo. Asimismo, para el caso del rectángulo, donde 13 imágenes son correctas, 2 consideradas como clavo y 5 como cuadrado. Mientras que, para la clase tuerca, 19 fueron clasificadas correctamente y solo uno considerado como un cuadrado. Lo que indica que existe una baja precisión y sensibilidad porque no se identifican las clases correctamente.

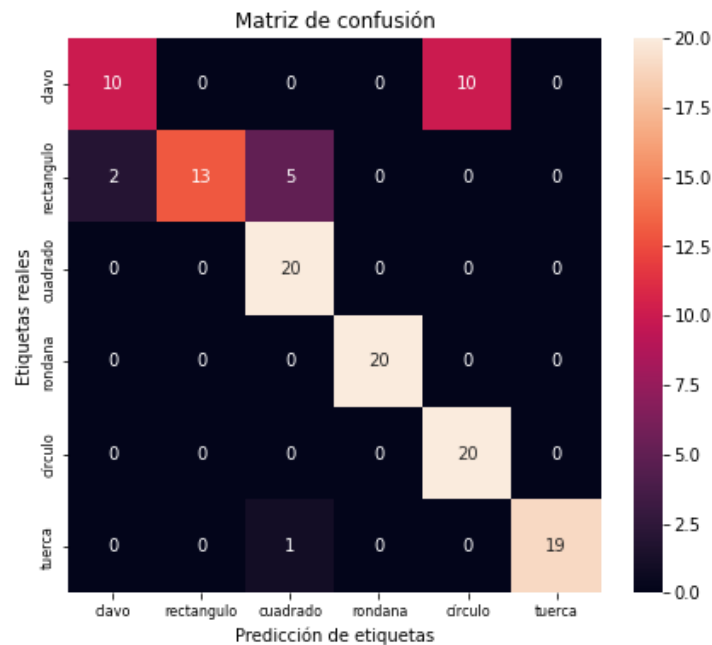


Figura 67. Matriz de confusión sin PCA con imágenes de 1.3 MP.

De acuerdo con la ejecución de combinaciones del diseño factorial, para imágenes de 1.3 MP, la combinación de 64 neuronas en capa oculta, optimizador SGD, imágenes de 1500

1x y 25 componentes, se alcanza una exactitud de 97.22 %, siendo la mejor combinación con el algoritmo RNC+PCA.

Con el algoritmo de RNC+PCA, la matriz de confusión se muestra en la Figura 68, donde para la clase clavo, 20 fueron correctas, para el rectángulo 17 correctas, de las 3 que faltan, 2 se clasificaron como cuadrado y 1 para rondana. Para las clases cuadrado, rondana y tuerca, las 20 etiquetas por clase fueron correctamente clasificadas. Lo que indica que detecta el 96.83 % de exactitud del total de las imágenes.

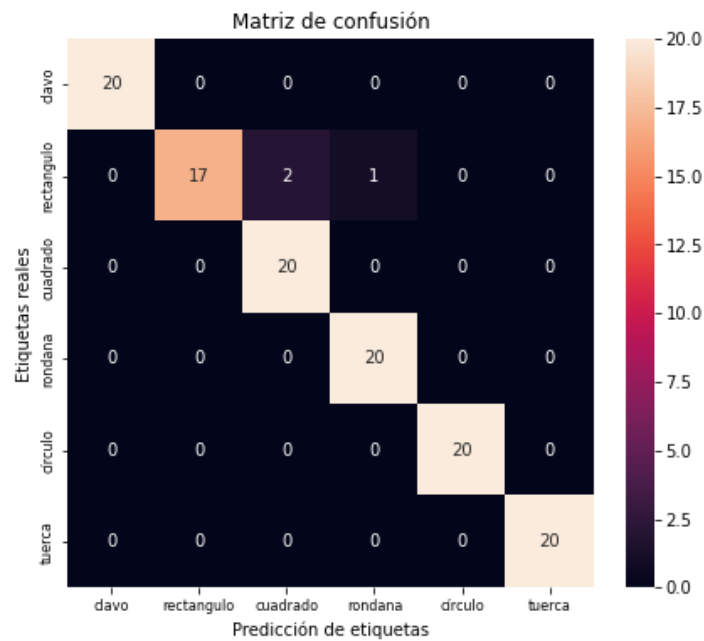


Figura 68. Ejemplo de matriz de confusión con imágenes de 1.3 MP.

Por lo tanto, los resultados de la matriz de confusión coinciden con el valor promedio de la exactitud del 96.83 % (resultado de la Tabla 12) en clasificación de imágenes de 1500 1x con 25 componentes y optimizador SGD, puesto que, de las 120 imágenes, solo 6 imágenes consideradas como clasificadas incorrectamente que equivale al 5 % de error.

La probabilidad de que las siguientes iteraciones sean confiables se determina a través de la evaluación del nivel de confianza para asegurar la capacidad de la RNC para clasificar. A partir del total de muestras de las iteraciones anteriores (con y sin PCA), se obtuvieron en total 500 muestras (cantidades), correspondiendo a los resultados de cada una de las métricas. De los datos se obtuvo la desviación estándar y el promedio, del total de

mediciones, obteniendo el valor de confianza del 95 % por las 1080 imágenes que se tienen en el *dataset*.

Para el caso del algoritmo sin PCA, los valores de las métricas corresponden como mínimo 76.27 % obtenido en F1 y como máximo 79.85 % en precisión, indicando la confianza de un 76 % aproximadamente, mientras que con el algoritmo de PCA con 25 componentes, el promedio de las métricas es del 95.80 % de precisión como valor máximo y como valor mínimo 95.22 % en sensibilidad, lo cual indica que el algoritmo con PCA es capaz de identificar las clases y las imágenes con un valor de confianza del 95 %.

*Tabla 13. Intervalos de confianza, con un tamaño de muestra de 500, en la validación.*

Datos	Exactitud		Sensibilidad		Precisión		F1	
	Sin PCA (%)	Con PCA (%)	Sin PCA (%)	Con PCA (%)	Sin PCA (%)	Con PCA (%)	Sin PCA (%)	Con PCA (%)
<b>Media</b>	78.89	95.75	76.66	95.22	79.85	95.80	76.27	95.50
<b>Desviación estándar</b>	8.11	2.46	8.85	4.14	8.45	4.27	8.83	3.78
<b>Promedio de confianza</b>	95.00	95.00	95.00	95.00	95.00	95.00	95.00	95.00
<b>Intervalo de confianza (izquierda)</b>	78.18	95.53	75.88	94.88	79.11	95.46	75.50	95.19
<b>Intervalo de confianza (derecha)</b>	79.60	95.96	77.43	95.55	80.59	96.15	77.05	95.80.

A partir del intervalo de confianza y de los valores de dispersión, se descarta el uso del algoritmo de la red neuronal sin PCA y se considera el algoritmo con PCA para realizar una interfaz para el uso del sistema.

## 6.6 Interfaz para el manejo del sistema

A partir de la red neuronal realizada en *Google Colab*, se crean dos archivos, donde uno corresponde a pesos y el otro la RNC, ambos con una extensión “.h5”, para utilizar la RNC de manera local y solo cargarlos en el archivo de interfaz destinado a realizar las predicciones correspondientes.

El sistema fue desarrollado en lenguaje Python, por lo que es multiplataforma (Windows, Mac, Linux y entre otros). Para utilizar el sistema es necesario tener conectada la cámara USB a una PC o laptop, de lo contrario no se podrá usar la función de clasificación en

tiempo real. Los archivos que utiliza el sistema están contenidos en una carpeta tal como se indica en la Figura 69.



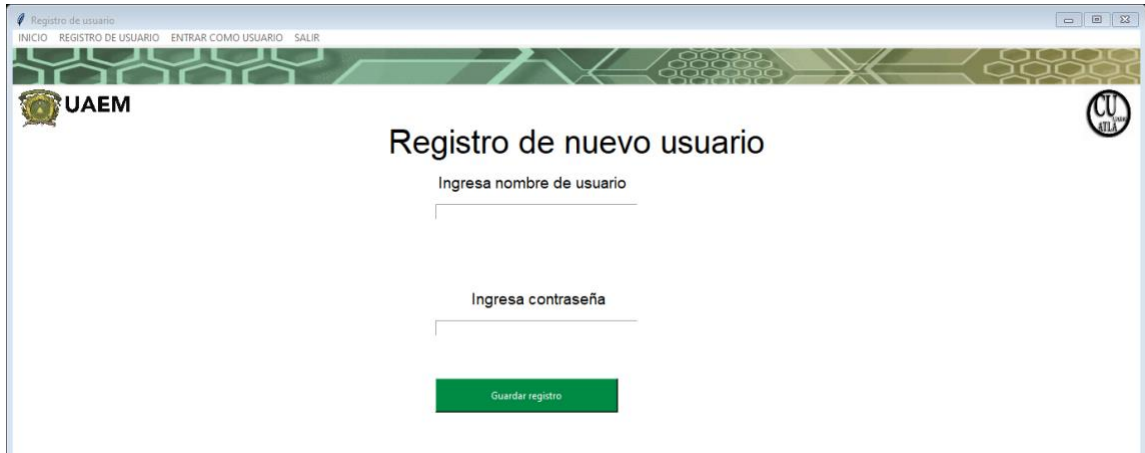
Figura 69. Archivos del sistema.

Como primer formulario que corresponde al inicio (Figura 70), se encuentra el menú con las opciones que podrá elegir el usuario, tales como: registrar un nuevo usuario, iniciar sesión y un menú para seleccionar entre clasificación en tiempo real o clasificación general de imágenes contenidas en una carpeta.



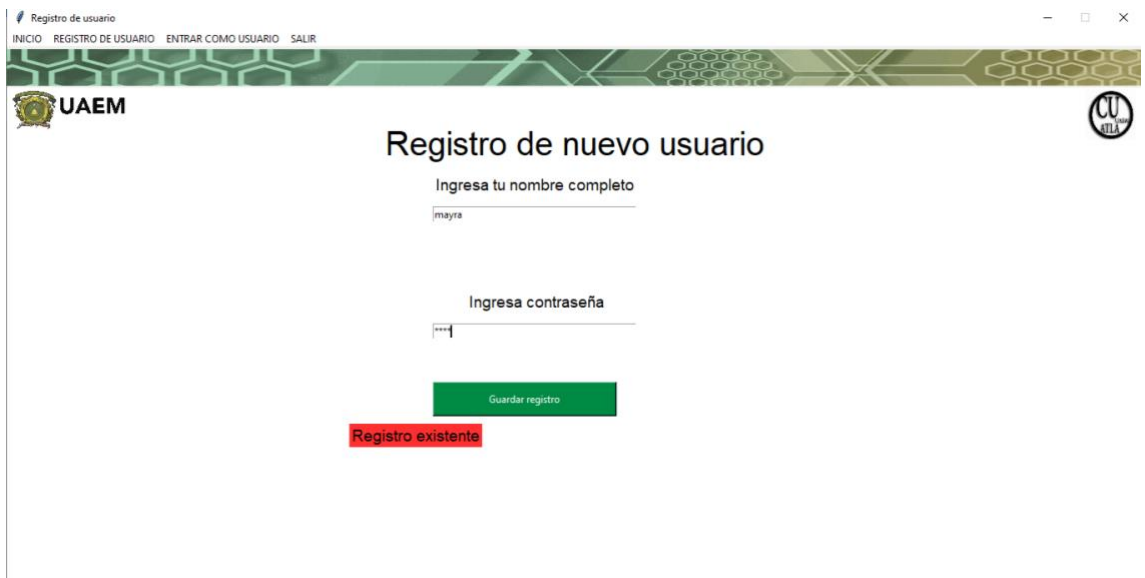
Figura 70. Menú de inicio.

Una de las funciones en el menú de inicio que se encuentra en la barra en la barra superior izquierda (Figura 70), es *REGISTRO DE USUARIO*, el cual, al darle clic a la opción, se visualiza como en la Figura 71.



*Figura 71. Registro de usuario.*

Al ingresar datos de un usuario ya registrado (Figura 72), se presenta un mensaje subrayado en color rojo, que indica que el registro existe, por lo tanto, se tiene que realizar únicamente un nuevo registro.



*Figura 72. Registro existente.*

Al registrar datos de un nuevo usuario (Figura 73) y al presionar el botón de guardar registro, se observan los datos almacenados, con los cuales podrá iniciar sesión.

*Figura 73. Registro de nuevo usuario.*

Los datos son almacenados en un archivo con extensión *.txt*, con el título “*Alta\_usuario*”. En la primera columna se muestra la *ID*, otorgada a cada uno de los usuarios, para llevar un control de registros mediante un índice. En la siguiente columna se observa el nombre y apellidos, y en la última columna se encuentra la contraseña (Figura 74).

Archivo	Edición	Formato	Ver	Ayuda
1	jose	1234		
2	mayramateo		1234	
3	valeria	1234		
4	mateo	1234		
5	paulino	1234		

*Figura 74. Almacenamiento de datos de registro de usuario.*

Una vez registrado el usuario, se puede tener acceso al sistema, mediante la opción “*ENTRAR COMO USUARIO*” (Figura 75), donde se coloca el nombre que se registró y la contraseña.



Acceso del usuario

INICIO REGISTRO DE USUARIO ENTRAR COMO USUARIO SALIR

UAEM CUATLA

### Acceso al sistema

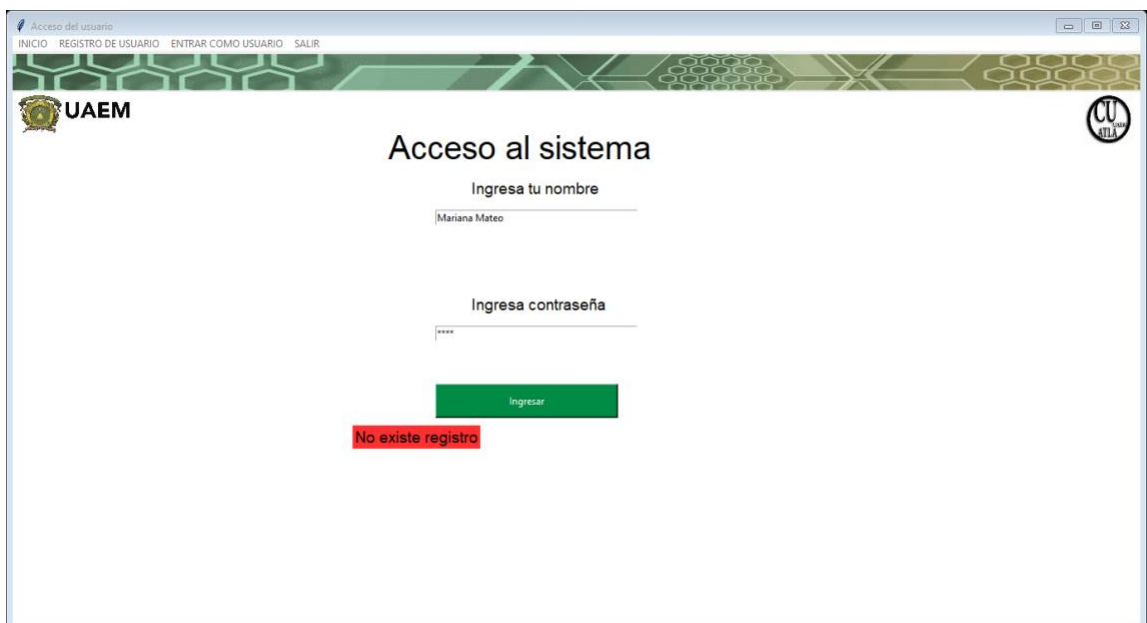
Ingresa tu nombre

Ingresa contraseña

Ingresar

*Figura 75. Entrar como usuario.*

Si la contraseña o/y el usuario no son correctos, se despliega el mensaje en pantalla de registro no existente (Figura 76).



Acceso del usuario

INICIO REGISTRO DE USUARIO ENTRAR COMO USUARIO SALIR

UAEM CUATLA

### Acceso al sistema

Ingresa tu nombre

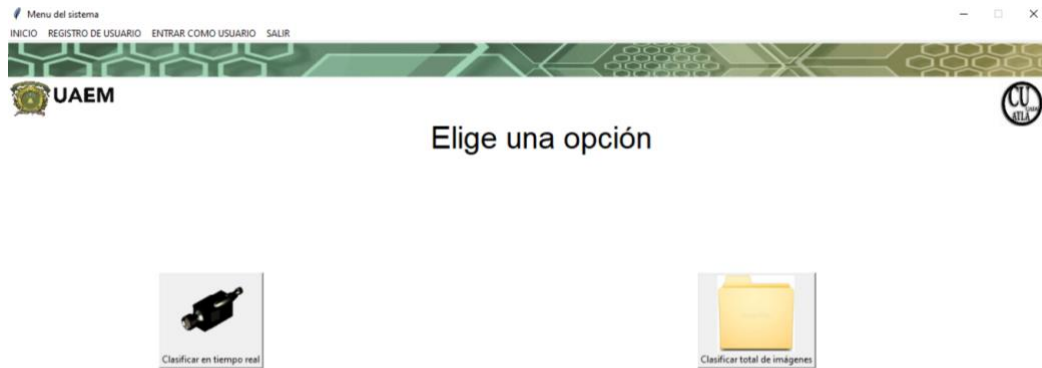
Ingresa contraseña

Ingresar

No existe registro

*Figura 76. Mensaje de registro no encontrado.*

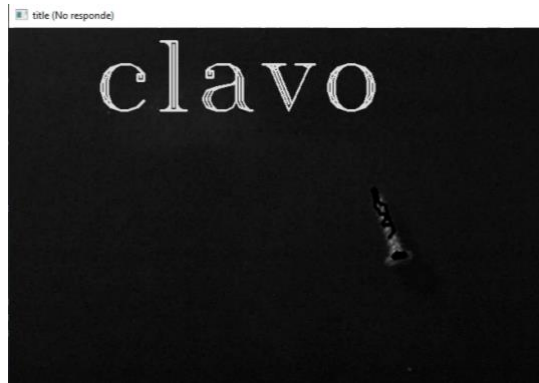
Si el usuario es correcto, entonces se aparece el menú para realizar alguna función del sistema (Figura 77), ya sea clasificar en tiempo real, clasificar total de imágenes, contenidas en una carpeta.



*Figura 77. Menú del sistema.*

Para elegir la opción de “*Clasificar en tiempo real*”, se debe tener conectada la cámara de 1.3 MP, de lo contrario se produce error al utilizar la opción puesto que no se encuentra la conexión USB activa de la cámara. Una vez conectada la cámara y dándole clic a la opción, se despliega la ventana de la cámara para poder clasificar las piezas metálicas con la forma regular (rectángulo, círculo, rondana, cuadrado) o irregular (clavo, tuerca). El ejemplo de uso se muestra en figura 69 y 70. En la figura 69, se clasificó un tornillo, que, por sus características, la RNC lo clasificó como un clavo, por lo tanto, se considera como un verdadero positivo:





*Figura 78. Clasificación de clavo.*

Como ejemplo de clasificación de rectángulo, se tiene la Figura 79, donde se identifica, con coordenadas los bordes del rectángulo para verificar que los lados de la pieza sean correctamente identificados para el recorte de la figura y posteriormente sea clasificada.



*Figura 79. Clasificación de rectángulo.*

Para esta sección, las imágenes clasificadas se almacenan en una carpeta para clasificarlas por la RNC, obteniendo como resultado la Figura 80:



*Figura 80. Clasificación de imágenes en carpeta.*

Además, el resultado de clasificación es almacenado como figura con extensión “.jpg”, para que el usuario pueda volver a consultarla. Un ejemplo del repertorio de imágenes almacenadas es el de la Figura 81.

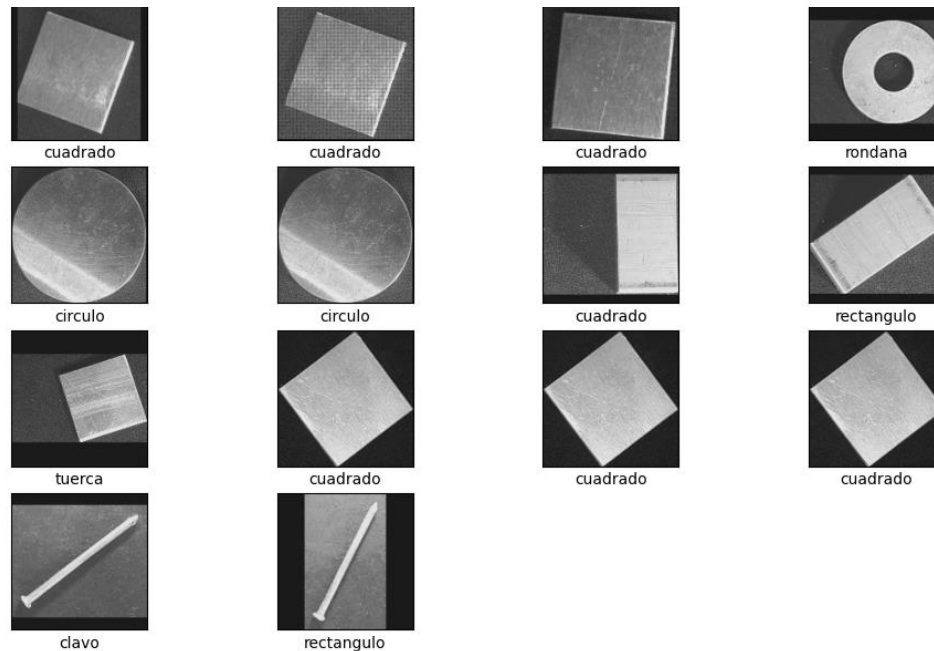


Figura 81. Resultados de clasificación (2).

De los resultados de clasificación se pueden obtener los datos estadísticos, que corresponde a la creación de la matriz de confusión y de la obtención de los porcentajes de las métricas: Precisión, Exactitud, Sensibilidad y F1. Cuando se está creando la matriz de confusión se despliega en pantalla un mensaje con el aviso de “Creando matriz de confusión” (Figura 82). Posteriormente, se muestra otro formulario con la matriz de confusión y los datos de las métricas (Figura 83).

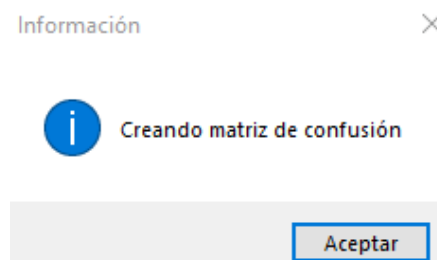


Figura 82. Aviso de espera para generación del resultado estadístico.

## Resultados de Clasificación

### Resultado de métricas

Exactitud : 93.75  
 Precisión : 94.29  
 Sensibilidad : 93.75  
 F1 : 93.70  
 Total de imágenes :48.00

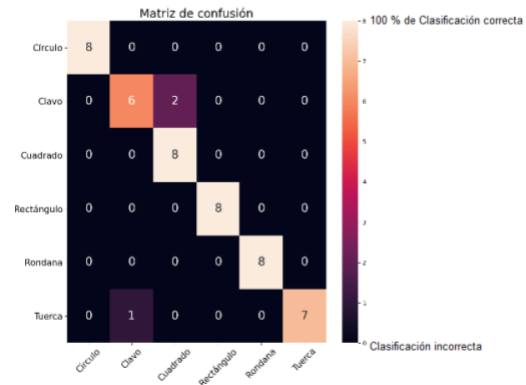


Figura 83. Matriz de confusión con resultados de métricas.

Por lo tanto, el desarrollo de la interfaz para el usuario está destinada para el uso de dos opciones, ya sea en tiempo real o clasificar imágenes que se encuentren en carpeta, luego de seleccionar cualquier opción, el usuario puede salir mediante la opción salir que se encuentra en el formulario. La interfaz es parte importante del trabajo, no solo para darle uso al sistema, sino para que el usuario conozca el resultado de las métricas, en el caso de los resultados estadísticos, para valorar el funcionamiento de la red neuronal convolucional mediante la cámara de 1.3 MP. Para el caso de la opción de clasificar en tiempo real, se comprueba el funcionamiento ante el uso de la iluminación a 1500 lx en las figuras regulares e irregulares.

## 7 CONCLUSIONES

En este trabajo se desarrolló un sistema de visión artificial para la clasificación de piezas metálicas mediante el algoritmo de Análisis de Componentes Principales y una Red Neuronal Convolucional, considerando las variables: iluminación, número de neuronas en capa oculta, tipos de optimizador y número de componentes PCA.

A partir de los resultados del diseño factorial  $3^4$  se realizan las pruebas con la cámara de 8 MP, el mejor resultado de exactitud en este caso le corresponde al 83.89 %, con la combinación de 32 neuronas, 25 componentes, optimizador SGD y el nivel de iluminación de 2000 lx, con luz cálida. Para el caso de las pruebas con la cámara de 1.3 MP expuestas a 1500 lx, con luz cálida, el mejor resultado de exactitud obtenido es de 97.22 % que corresponde a la combinación de 25 componentes, 64 neuronas, optimizador SGD. De las dos cámaras, el mejor porcentaje de exactitud se obtuvo con la cámara de 1.3 MP, ya que en las imágenes se identifica el contorno de los objetos y las características como líneas dentro de las figuras no se perciben de manera detallada, lo cual ayuda a la identificación correcta de la pieza metálica, además la iluminación es uniforme en comparación con las imágenes obtenidas de 8MP. En estas, la reflexión de la luz sobre la superficie de la pieza es más notoria y ocasiona que se deforme visualmente la forma geométrica de la pieza.

La exposición de las piezas metálicas a diferentes escalas de iluminación (1000, 1500 y 2000 luxes), hace que el tono de fondo sea similar al tono del borde de la pieza, generando tanto falsos positivos como negativos y, en consecuencia, no se logra la correcta identificación. Para reducir este problema, se utilizó un algoritmo para modificar el balance de iluminación y reemplazar el fondo por una máscara de tonalidad negra uniforme (valor en RGB de 0,0,0). De esta manera se logró resaltar el borde de la pieza metálica.

Para que el algoritmo de la red neuronal procese los datos de las imágenes adquiridas en 2 dimensiones, es necesario que el largo y ancho de estas tengan el mismo número de píxeles, por lo que se desarrolló una etapa dentro del algoritmo de pretratamiento para lograr este proceso y redimensionar las imágenes a un tamaño de 150 x 150 píxeles.

En las pruebas donde se expusieron las piezas a luz cálida y fría con intensidad de 1000 lx, se obtuvieron porcentajes, como valor máximo 95.56 % y como mínimo 70.83 % para la métrica de exactitud, obteniendo el 29.77 % de falsos positivos y negativos. Lo anterior se debe a que no se reconocen los bordes de las imágenes de manera adecuada, debido a la opacidad de estas. Por lo tanto, se concluye que a 1000 luxes, independientemente del tipo de iluminación, la identificación de las piezas no se considera adecuada.

En el caso de las pruebas realizadas con 2000 lx, se obtuvo para la métrica de exactitud un valor máximo de 93.83 % y 72.17 % como mínimo, los falsos positivos fueron del 27.83 %. Lo anterior se debe a que por la cantidad de luz que incide en ellas, provoca que el fondo de la imagen se visualice como parte de la pieza metálica, especialmente en las piezas con las formas cuadrado, rectángulo y círculo. Por lo anterior, se considera que esta escala de iluminación tampoco es adecuada para la clasificación de las piezas metálicas.

Para elegir el número de componentes se consideró la cantidad de varianza acumulada junto con la desviación estándar que generan el conjunto de muestras de exactitud de las clasificaciones realizadas en el apartado de método, para evaluar la dispersión de los datos. De los resultados obtenidos en el promedio de las tres iteraciones, la mejor combinación fue con 25 componentes, optimizador SGD, 64 neuronas en capa oculta, imágenes de 1.3 MP expuestas a 1500 lx, varianza acumulada del 90 %, con una desviación estándar del 2.40 %, lo cual permitió eliminar información redundante, dejando solo la que proporciona importancia en crear las imágenes. También el promedio del porcentaje de exactitud aumentó del 85 % (valor mínimo) al 95.75 % (valor máximo). Por lo tanto, entre más alto sea el porcentaje de desviación estándar, el resultado de exactitud está más alejado del valor promedio, pero en el intervalo del valor mínimo y máximo. El patrón de clasificación se basa en los contornos geométricos de las figuras, para ello es importante detectar los bordes de las imágenes y realizar el balance de iluminación e identificar correctamente el contorno. La variación que existe en la detección de contornos provoca la determinación de falsos negativos y positivos.

El uso de PCA para el pretratamiento de imágenes de piezas metálicas con cámara de 1.3 MP expuestas a 1500 lx en una red neuronal convolucional se obtiene de la clasificación del diseño factorial el 97.22 % de exactitud y en el análisis que se realizó del promedio de

500 muestras, los resultados corresponden a: Exactitud 95.75 %, Sensibilidad 95.22 %, Precisión 95.80 %, F1 95.50 %. Por lo tanto, en el diseño factorial y en análisis de las 500 muestras, el porcentaje establecido en la hipótesis del 93 % para la métrica de exactitud fue superado.

El resultado de la clasificación de la Red Neuronal Convolutiva sin el algoritmo PCA fue del 85 % para la métrica de exactitud, mientras que aplicando la red con el algoritmo de PCA se obtuvo 95.75 % para la misma métrica. Al comparar estos resultados se observó una mejora del 10.75 % en la métrica de exactitud, por lo tanto, la Red Neuronal Convolutiva aplicada con el algoritmo de PCA, mejora el proceso de clasificación de imágenes metálicas.

### **Productos obtenidos y trabajo a futuro**

Los productos obtenidos durante el desarrollo del proyecto es la publicación del artículo *Analysis and comparison of a classification of regular metal pieces by Convolutional Neural Networks with and without Principal Component Analysis*, enviado a CORE en la revista *Journal Research in Computing Science* y el desarrollo de la interfaz del sistema para la clasificación de piezas metálicas.

Como trabajo a futuro se considera incrementar la cantidad de imágenes tanto de entrenamiento como para validación, respecto al 60 % para entrenamiento y 40 % para validación, con el motivo de extender las características que proporcionan las piezas metálicas a la red neuronal convolutiva y evaluar si mejora el porcentaje de las métricas, además incluir otro tipo de piezas metálicas, que incluyan colores diferentes a plateado o dorado para verificar el funcionamiento de la red neuronal y evaluar el porcentaje de exactitud que se obtenga.

De acuerdo con algoritmo de PCA, en caso de incrementar imágenes, realizar el análisis para valores cercanos a 25 componentes ya que presenta menor cantidad de desviación estándar con el 2.40 % y a partir de esto, determinar cuánto mejoran las métricas propuestas al considerar componentes que proporcionen una desviación estándar alrededor del 2.40 % y varianza aproximada al 90 % o realizar un algoritmo para determinar la cantidad de componentes adecuado. Para la interfaz, la cantidad de imágenes

a clasificar que están almacenadas en carpeta está delimitada a 48 imágenes, por lo tanto, se puede mejorar ese aspecto para que el usuario defina cuantas imágenes clasificar.

## 8 REFERENCIAS

Aguilar Domínguez, K., Mejía Lavalle, M., y Sossa, H. (2021). Mejora eficiente de la luminosidad en imágenes del cerebro humano utilizando redes neuronales pulso acopladas. *Computación y Sistemas*, 1(24), 105-120.

Aguilar Torres, E., y Bekios Calfa, J. (2016). Clasificación de género utilizando vectores de frecuencia basados en descriptores locales. *Revista chilena de ingeniería*, 24(1), 124-134.

Ali Ahmed, S., Dey, S., y Kumar Sarma, K. (2011). Image texture classification using artificial neural network. *Second national conference on emerging trends and applications in computer science*, 1-4.

Alsmadi, M. k., Omar, K. B., Noah, S. A., y Almarashdah, I. (2009). Performance comparison of multi layer perceptron (back propagation, delta rule and perceptron) algorithms in neural networks. *International Advance Computing Conference*, 1-4.

Ariza Lopez, F., Rodríguez Avi, J., y Alba Fernández, V. (2018). Control estricto de matrices de confusión por medio de distribuciones multinomiales. *GeoFocus. Revista internacional de ciencia y tecnología de la información geográfica*, 1(21), 1-13.

Aziz, L., Salam, H., Bin Haji Salam, S., Ullah Sheikh, U., y Ayub, S. (2020). Exploring Deep Learning-Based Architecture, Strategies applications and current trends in generic object detection: a comprehensive review. *Department of computer engineering*, 8(1), 1-35.

Basulto Rodríguez, Y. (2018). Integración de la red neuronal convolucional con el algoritmo de función de frontera de objeto para reconocimiento de piezas y detección de defectos. *Corporación mexicana de investigación en materiales . División de estudios de posgrado. Tesis para obtener el grado académico de maestro en ciencia y tecnología en manufactura avanzada*, 1-87.

Berzal, F. (2018). *Redes neuronales y deep learning* (Primera ed.). España: Editorial universidad de granada.

Borja Robalino, R., y Monleon Getino, A. (2020). Estandarización de métricas de rendimiento para clasificadores machine y deep learning. *Revista ibérica de sistemas y tecnologías de información*, 172-184.

Caballero Julián, F. G., Vidal Reyes, M., López Sánchez, A., y Jerónimo Ríos, C. A. (2017). Reconocimiento facial por el método de Eigenfaces. *Pistas Educativas*, 39(127), 1-16.

Calvete Sanz, D. (2018). Diseño, desarrollo y evaluación de un sistema de visión artificial para inspección de calidad a final de línea de fabricación del árbol de levas. *Universidad politécnica de Valencia*, 1-6.

Camacho, F., y Labrandero, J. (2016). Redes neuronales convolucionales aplicadas a la traducción del lenguaje verbal español al lenguaje de señas Boliviano. *Revista ciencia, tecnología e innovación*, 12(13), 1-8.

Cervantes Osornio, R., Arteaga Ramírez, R., Vázquez Peña, M. A., y Ojeda Bustamante, W. (2016). Red neuronal artificial backpropagation versus modelos empíricos para estimación de radiación global diaria en Sinaloa, México. *Revista mexicana de ciencias agrícolas*, 7(5), 1029-1042.

Chaides, O., Ahuett, H., Flores, A., Caballero, A., y Ruíz, L. (2009). Diseño y prueba de un sistema de control de espaciamiento y potencia para micro-EDM. *Centro de innovación en diseño y tecnología. Departamento de ingeniería mecánica, división de ingeniería y arquitectura*, 3(2), 1-10.

Cognex. (2020). *Introducción a la visión artificial* (Primera ed.). Estados Unidos: Cognex.

Cruz Roa, A., Suárez, A. S., Jiménez L, A. F., y Castro Franco, M. (2017). Classification and automatic mapping of land covers in satellite images using Convolutional Neural Networks. *Orinoquia, Universidad de los Llanos, Villavicencio, Meta*, 21(1), 1-13.



Delgado Rivero, F. (2001). Medición de eficiencia con redes neuronales artificiales. Una explicación al servicio de recogida de basuras. *Cuadernos de economía y dirección de la empresa*(25), 53-81.

Diario Oficial de la Federación. (2008). *Diario oficial de la federación*. Recuperado el 19 de 05 de 2021, de [http://dof.gob.mx/nota\\_detalle.php?codigo=5076393yfecha=30/12/2008](http://dof.gob.mx/nota_detalle.php?codigo=5076393yfecha=30/12/2008)

Díaz del Castillo Rodríguez, F. (2011). Procesos de maquinado sin arranque de viruta. *Facultad de estudios superiores Cuautitlán. Departamento de ingeniería.*, 1-69.

El Gamal, A., y Eltoukhy, H. (2005). CMOS image sensors. *Circuits and devices magazine* , 2(13), 6-20.

Esqueda Elizondo, J. J., y Palafox Maestre, L. E. (2005). *Fundamentos de procesamiento de imágenes*. Baja California, México.

European machine vision association. (2006). *BASLER*. Recuperado el 12 de 04 de 2021, de <https://www.baslerweb.com/en/>

Fernandez, M. E., Guzman, A., y Dadios, E. (2018). An artificial neural network model for the cell density measurement of spirulina (A. platensis). *Technological university of the Philippines*, 1-5.

Ferreira Mejía, P. A., Pérez Cubillos, W. N., y Aparicio Pico, L. E. (2021). Métodos de compresión sin pérdidas de imágenes de resonancia magnética utilizando transformada wavelet: revisión sistemática. *Modelo matemático*, 12(1), 54-56.

Filipo Rugeles, V. H., Cano Garzón, H. B., y Chavez Osorio, J. A. (2010). Aplicaciones de iluminación con LED. *Scientia et technica*, 16(45), 13-18.

Flores Eraña, J. A. (2009). Síntesis digital de color utilizando tonos de gris. *Tesis para obtener el grado de maestro en ciencias aplicadas*, 1-56.

Garro, B., Sossa, H., y Vázquez, R. (2012). Diseño automático de redes neuronales artificiales mediante el uso del algoritmo de evolución diferencial. *Centro de investigación en computación del instituto politécnico nacional*, 1.

Gazzaz, N., Yusoff, M. K., Aris, A. Z., Juahir, H., y Ramli, M. F. (2019). Artificial neural network modeling of the water quality index for Kinta River (Malaysia) using water quality variables as predictors. *Marine Pollution Bulletin*, 64(11), 2409-2420.

Gitesh Chitaliya, N., y Trivedi. (2010). Feature extraction using wavelet-PCA and neural network for application of object classification y face recognition. *Second international conference on computer engineering and applications*, 1, 1-5.

González Marcos, A., Martínez de Pisón Ascacibar, F. J., y Verónica Pernía, A. (2006). Técnicas y algoritmos básicos de visión artificial recurso electrónico. *Universidad de la rioja*, 50-97.

Guillermo González, I. (2014). Estudio de maquinabilidad de dos superaleaciones INCONEL 617 y 718. *Obtención de grado de doctor en ingeniería de materiales. Universidad autónoma de Nuevo León*, 220-223.

Ingrosuppliers. (2021). *Ficha técnica 381,382*. Recuperado el 25 de 02 de 2021, de <https://www.ingrosuppliers.com/product-page/uni-t-ut382-lux%C3%B3metro-hasta-20000lux-o-2000fc-con-interfaz-usb>

Islam, T., Rahman, S., Siddique Nafiz, K., y Jabid, T. (2018). Food image classification with convolutional neural network. *Artificial intelligent, robotics, and human computer interaction*, 2(1), 1-6.

Kalpakjian, S., y Steven, S. (2014). *Manufactura, ingeniería y tecnología* (Séptima ed.). México: Pearson.

Khan, S., Rahmani, H., Shah, S. A., y Bennamoun, M. (2018). A guide to convolutional neural networks for computer vision. *Morgan y Claypool*, 200-209.

Lira Chávez, J. (2018). *Tratamiento digital de imágenes multiespectrales* (3 ed.). México: UNAM.

Londoño Osorio, V., Marín Pineda, J., y Arango Zuluaga, E. (2013). Introducción a la visión artificial mediante prácticas de laboratorio diseñadas en matlab. *Tecnológicas. Instituto tecnológico metropolitano*, 591-603.

- Martín, O., López, M., y Martín, F. (2006). Redes neuronales artificiales para la predicción de la calidad en soldadura por resistencia por puntos. *Revista de metalurgia*, 5(42), 345-353.
- Martínez Susunaga, J. (2017). Sistema de visión artificial para la detección de espacio ocupado en lugares cerrados. *Trabajo final del proyecto de grado con el fin de obtener el título de Ingeniería en Telecomunicaciones. Bogotá D.C*, 80-97.
- Mathworks. (2021). *Mathworks*. Recuperado el 14 de 02 de 2021, de <https://es.mathworks.com/help/images/specify-adjustment-limits-as-range.html>
- MATLAB. (2021). *MATLAB*. Recuperado el 15 de 05 de 2021, de <https://n9.cl/a6p3k>
- Mazloom, M., y Kasaei, S. (2005). Face Recognition using Wavelet, PCA, and Neural Networks . *Proceeding of the First International Conference on Modeling, Simulation and Applied Optimization, Sharjah, U.A.E*, 1-6.
- Menéndez, A., Javier, R., López Portilla, V., Maykel, B., y Inglesias Martínez, M. E. (2013). Diseño de un sistema de transferencia y procesamiento de imágenes sobre un FPGA. *Revista de la Facultad de Ingeniería Universidad Central de Venezuela*, 2(28), 13-22.
- Mera Jiménez, L., y Ochoa Gómez, J. F. (2021). Redes neuronales convolucionales para la clasificación de componentes independientes de rs-fMRI. *TecnoLógicas*, 24(50), 19.
- Miranda, M. (2009). La imagen digital. *Gen*, 63(2), 134-136.
- Morejón Ortega, A. (2019). Sistema de Reconocimiento de Patrones de Imágenes e Inferencia de Acciones de un Brazo Robótico en un Ambiente no Estructurado Aplicado al Proceso de Almacenaje Robotizado. *Corporación Mexicana De Investigación en Materiales S.A. de C.V*, 1-99.
- Naveed Abbasi, A., y He, M. (2019). Convolutional neural network with PCA and batch normalization for hyperspectral image classification. *School of electronics and Information northwestern polytechnical university*, 959-962.

- Oludare Isaac, A., Jantan, A., y Dada, K. V. (2019). Comprehensive Review of Artificial Neural Network Applications to Pattern Recognition. *School of Computer Sciences, Universiti Sains Malaysia, Minden*, 7, 1-27.
- Pérez Lara, C., Ochoa Somuano, J., Pereyra Ramos, C., y Toscano Martinez, J. (2013). Reconocimiento de rostros en un ambiente de iluminación controlado. *X Congreso Internacional sobre Innovación y Desarrollo Tecnológico*, 1-6.
- Petatan Bahena, K. E., Gutiérrez Rojas, I., Abúndez Pliego, A., Mayén Chaires, J., y Andrés, B. O. (2021). Efecto de la velocidad de avance en torneado CNC en seco y el acabado superficial en la vida a fatiga de una aleación de aluminio 6061-T6. *Jornada de Ciencia y Tecnología Aplicada*, 4(1), 235-240.
- Rendón Macías, M. E., Villasís Keeve, M. Á., y Miranda Novales, M. G. (2016). Estadística descriptiva. *Revista Alergia México*, 63(4), 397-407.
- Resendiz Tolentino, G. J. (2017). Filtros Wavelet óptimos para detección de ruido en imágenes SD y HD. *Universidad Nacional Autónoma de México. Facultad de Ingeniería.*, 1-111.
- Reyes Nava, A. (2018). Clasificación de factores de riesgo de alumnos basado en Redes Neuronales Artificiales y Minería de Datos. *Tesis que para obtener el Grado Académico de: Maestro(a) en Ciencias de la Computación en el Centro universitario UAEM Atlacomulco*, 1-87.
- Rivas Asanza, W., y Bertha, M. O. (2018). Redes neuronales artificiales aplicadas al reconocimiento de patrones. *Ediciones UTMACH*, 1-29.
- Riveros Guevara, A., Salas López, C. N., y Solaque Guzmán, L. (2012). Aproximación a la navegación autónoma de una plataforma móvil , mediante visión estereoscópica artificial . *Ciencia e ingeniería neogranadina*, 22(2), 111 - 129.
- Rocha Muñoz, T., y Ortiz de Lazcano Lobato, J. M. (2016). Sistema de recomendación basado en redes neuronales competitivas. *Escuela Técnica Superior de Ingeniería Informática*, 1-47.

Rojas Hernández, R., Silva Ortigoza, R., y Molina Vilchis, M. A. (2007). La Visión Artificial en la Robótica. *Polibits*(35), 22-28.

Rosas Echevarria, C. W., Solis Bonifacio, H., y Cueva Cerna, A. F. (2019). Sistema eficiente y de bajo costo para la selección de granos de café: una aplicación de la visión artificial. *Scientia Agropecuaria*, 10(3), 347-351.

Salazar Cárdenas, D., y Vargas Trujillo, I. A. (2017). Diseño e implementación de un sistema creador de arte digital basado en el concepto de "cadáver exquisito" utilizando inteligencia artificial. *Universidad Autónoma de Occidente. Facultad de Ingeniería. Departamento de Automática y Electrónica. Programa Ingeniería Mecatronica*, 1-102.

Salgado Díaz del Río, A. (2019). Reconocimiento automático de instrumentos mediante aprendizaje máquina. *Trabajo fin de grado de ingeniería de tecnologías de telecomunicación. Teoría de la señal y comunicaciones escuela técnica superior de ingeniería*, 1-74.

Sánchez Reinoso, J. S. (2011). *Identificación de objetos en tiempo real utilizando técnicas y clasificadores de visión artificial para el reconocimiento de patrones* . Ecuador.

Sánchez, G. (2018). Planificación e implementación del análisis de los sistemas de medición en la empresa Denso Manufacturing, bajo los lineamientos del Manual MSA. *Facultad de Ciencias Exactas, Físicas y Naturales. Escuela de Ingeniería Industrial*, 1-106.

Sánchez, J. A. (2006). *Instrumentación y control básico de procesos* (Primera ed.). España: Díaz de Santos.

Santamaría Colula, L. Á. (2019). Reconocimiento de patrones en imágenes médicas por medio de redes neuronales convolucionales. *Tesis para obtener el título de Maestro en ciencias de la computación. Benemérita Universidad Autónoma de Puebla*, 1-76.

Scherer, D., Müller, A., y Behnke, S. (2010). Evaluation of Pooling Operations in Convolutional Architectures for Object Recognition. *University of Bonn, Institute of Computer Science VI, Autonomous Intelligent Systems Group, Romerstr*, 1-10.

Soares, F., y Souza, A. (2016). *Neural network programming whit java* (1 ed.). Birmingham: Packt publishing.

Tiwari, S. (2019). *Complete guide to machine learning evaluation metrics*. Recuperado el 16 de 11 de 2020, de <https://medium.com/analytics-vidhya/complete-guide-to-machine-learning-evaluation-metrics-615c2864d916>

Utrilla Diaz, M., y Reyes Hernández, A. (2019). Diseño de un sistema de indicadores de procesos para el maquinado de piezas en el torno convencional del ITSS. *Obtención de grado de maestro en manufactura avanzada*, 1-109.

Xatakandroid. (07 de 02 de 2019). *Xatakandroid*. Recuperado el 12 de 04 de 2021, de <https://www.xatakandroid.com/moviles-android/moto-g7-play-caracteristicas-precio-ficha-tecnica>

Yang, Q. (1993). Classification of apple surface features using machine vision and neural networks. *Computers and Electronics in Agriculture*, 1-12.

Younas, T., Kumari, J., y Manzoor, M. (2017). Non conventional machining processes as expedient alternatives for conventional machining processes. *International conference on engineering technologies and social sciences*, 5.

Zambrano Escobar, A. P. (2009). Aplicación de las redes neuronales artificiales en procesadores digitales de señales: caracterización de sensores infrarrojos. *Centro de Investigación de las redes neuronales artificiales y la robótica en la universidad nacional experimental politécnica*, 13(51), 129-136.

## 9 ANEXOS

### Resultados del diseño factorial, prueba 1.

Resultados de la prueba 1 (Tabla 14-Tabla 16). Se presenta el total de combinaciones. La tabla contiene el número de capas ocultas: 16, 32 y 64, tipo de optimizador: SGD, RmsProp y Adam, nivel de iluminación: 1000, 1500 y 2000 lx, número de componentes: 2, 25 y 50. Cada una de las combinaciones generó un porcentaje de exactitud de la clasificación de las imágenes de 1.3MP y de 8MP.

*Tabla 14. Resultados de la prueba 1 del diseño factorial 3<sup>4</sup>.*

PRUEBA 1					
Numero de capas ocultas	Optimizador	Iluminación	Componentes	Cámara	
				1.3M	8M
				Exactitud	Exactitud
16	SGD	1500	25	85.83%	79.17%
64	RmsProp	1500	2	92.60%	78.33%
16	RmsProp	1500	25	94.17%	75.00%
16	RmsProp	1000	50	93.33%	75.00%
16	SGD	2000	2	93.33%	68.60%
16	SGD	1000	50	94.17%	65.83%
32	RmsProp	1500	50	93.33%	80.00%
32	SGD	1000	2	85.83%	70.83%
64	SGD	2000	50	94.17%	51.24%
16	SGD	2000	50	94.17%	78.51%
32	Adam	1000	50	94.17%	70.00%
32	RmsProp	1000	25	95.83%	70.83%
16	Adam	1500	50	99.17%	80.83%
16	RmsProp	1000	25	95.83%	67.50%
16	Adam	1500	2	88.33%	80.00%
32	Adam	1000	25	94.17%	70.83%
16	Adam	1000	50	94.17%	70.83%
32	SGD	2000	25	94.17%	49.59%
64	SGD	1500	2	95.41%	77.50%
32	RmsProp	1000	50	93.33%	67.50%
16	SGD	1000	25	96.67%	65.83%
16	RmsProp	1500	50	97.51%	77.50%

Tabla 15. Resultados de la prueba 1 del diseño factorial 3<sup>4</sup> (2).

32	Adam	2000	25	94.17%	80.99%
32	Adam	1000	2	95.83%	70.83%
64	Adam	2000	2	95.00%	68.60%
16	Adam	2000	2	92.50%	49.59%
64	Adam	2000	50	89.17%	75.21%
64	RmsProp	1000	50	95.83%	69.17%
32	SGD	2000	50	93.33%	50.41%
32	RmsProp	2000	25	91.67%	78.51%
64	SGD	2000	2	94.17%	70.25%
32	Adam	1500	50	88.33%	80.00%
64	Adam	1500	50	97.52%	77.50%
32	RmsProp	2000	2	90.00%	72.50%
32	RmsProp	1000	2	94.17%	68.33%
16	Adam	1000	2	93.33%	70.00%
32	RmsProp	2000	50	92.50%	76.03%
32	SGD	1500	50	94.17%	80.00%
32	Adam	2000	50	92.50%	80.00%
16	Adam	1500	25	98.33%	79.17%
16	Adam	1000	25	97.25%	70.83%
32	RmsProp	1500	25	95.17%	76.67%
32	SGD	1000	50	93.33%	72.50%
16	RmsProp	2000	50	93.33%	79.17%
64	Adam	1000	2	87.50%	70.00%
64	SGD	1500	25	98.33%	80.00%
32	RmsProp	1500	2	88.33%	75.83%
16	RmsProp	2000	25	94.17%	72.50%
64	RmsProp	2000	50	97.50%	78.51%
64	Adam	2000	25	94.17%	64.46%
16	SGD	1000	2	86.67%	70.83%
16	RmsProp	2000	2	90.83%	84.30%
64	SGD	1500	50	89.17%	77.50%
64	SGD	1000	2	93.33%	75.00%
64	RmsProp	1500	25	97.53%	76.67%
64	SGD	2000	25	94.17%	81.41%
64	SGD	1000	50	85.00%	70.00%
64	Adam	1500	25	97.52%	79.17%
32	SGD	1500	2	96.67%	77.50%
16	RmsProp	1500	2	88.33%	75.00%



Tabla 16. Resultados de la prueba 1 del diseño factorial  $3^4$  (3)

64	RmsProp	2000	2	94.17%	52.89%
64	RmsProp	1500	50	93.33%	70.83%
64	Adam	1000	50	92.50%	75.00%
64	RmsProp	1000	25	95.00%	74.17%
64	Adam	1000	25	95.00%	70.83%
32	Adam	2000	2	92.64%	83.33%
64	RmsProp	2000	25	84.17%	69.17%
16	SGD	1500	2	97.52%	78.33%
64	SGD	1000	25	93.33%	66.67%
16	SGD	1500	50	94.17%	80.00%
32	Adam	1500	25	93.33%	78.33%
16	SGD	2000	25	92.50%	77.69%
16	RmsProp	1000	2	95.00%	72.50%
16	Adam	2000	50	90.83%	50.41%
32	Adam	1500	2	94.17%	78.33%
64	RmsProp	1000	2	86.67%	75.00%
32	SGD	1500	25	96.67%	75.83%
32	SGD	2000	2	94.17%	52.07%
64	Adam	1500	2	96.67%	78.33%
32	SGD	1000	25	95.83%	70.00%
16	Adam	2000	25	93.33%	49.59%

Resultados de la prueba 2 (Tabla 17-Tabla 19 ).

Se utilizaron las mismas variables, únicamente se repitieron las mismas condiciones para evaluar el porcentaje de exactitud.

Tabla 17. Resultados de prueba 2 del diseño factorial  $3^4$ .

PRUEBA 2					
Numero de capas ocultas	Optimizador	Iluminación	Componentes	Cámara	
				1.3M	8M
				Exactitud	Exactitud
16	SGD	1500	25	92.50%	79.17%
64	RmsProp	1500	2	82.50%	75.35%
16	RmsProp	1500	25	93.33%	75.12%
16	RmsProp	1000	50	94.17%	67.50%

Tabla 18. Resultados de la prueba 2 del diseño factorial  $3^4$  (2)

16	SGD	2000	2	94.17%	80.99%
16	SGD	1000	50	94.17%	70.83%
32	RmsProp	1500	50	96.67%	75.17%
32	SGD	1000	2	93.33%	69.17%
64	SGD	2000	50	94.17%	76.86%
16	SGD	2000	50	94.17%	78.51%
32	Adam	1000	50	95.83%	67.50%
32	RmsProp	1000	25	94.17%	76.67%
16	Adam	1500	50	90.00%	79.17%
16	RmsProp	1000	25	85.83%	70.83%
16	Adam	1500	2	95.00%	81.67%
32	Adam	1000	25	92.50%	70.00%
16	Adam	1000	50	94.17%	70.83%
32	SGD	2000	25	92.50%	49.59%
64	SGD	1500	2	95.83%	77.92%
32	RmsProp	1000	50	85.00%	70.83%
16	SGD	1000	25	95.00%	67.50%
16	RmsProp	1500	50	95.83%	79.17%
32	Adam	2000	25	94.17%	76.03%
32	Adam	1000	2	95.83%	75.00%
64	Adam	2000	2	90.83%	49.59%
16	Adam	2000	2	94.17%	50.41%
64	Adam	2000	50	90.83%	49.59%
64	RmsProp	1000	50	94.17%	70.83%
32	SGD	2000	50	93.33%	82.64%
32	RmsProp	2000	25	92.50%	78.51%
64	SGD	2000	2	80.00%	69.42%
32	Adam	1500	50	96.67%	76.67%
64	Adam	1500	50	95.00%	76.67%
32	RmsProp	2000	2	94.17%	75.00%
32	RmsProp	1000	2	95.00%	72.50%
16	Adam	1000	2	93.33%	77.50%
32	RmsProp	2000	50	91.67%	76.86%
32	SGD	1500	50	97.50%	81.58%
32	Adam	2000	50	93.33%	49.59%
16	Adam	1500	25	97.50%	79.17%
16	Adam	1000	25	96.50%	70.83%
32	RmsProp	1500	25	85.83%	76.67%
32	SGD	1000	50	93.33%	70.83%
16	RmsProp	2000	50	92.50%	73.33%

Tabla 19. Resultados de la prueba 2 del diseño factorial 3<sup>4</sup> (3)

64	Adam	1000	2	96.67%	73.33%
64	SGD	1500	25	97.50%	79.17%
32	RmsProp	1500	2	85.00%	75.83%
16	RmsProp	2000	25	97.50%	72.50%
64	RmsProp	2000	50	94.17%	77.69%
64	Adam	2000	25	92.50%	65.29%
16	SGD	1000	2	95.83%	71.67%
16	RmsProp	2000	2	93.33%	81.82%
64	SGD	1500	50	98.33%	77.54%
64	SGD	1000	2	95.00%	70.83%
64	RmsProp	1500	25	87.50%	78.33%
64	SGD	2000	25	92.50%	81.82%
64	SGD	1000	50	85.00%	68.33%
64	Adam	1500	25	98.33%	75.18%
32	SGD	1500	2	97.50%	76.67%
16	RmsProp	1500	2	87.50%	75.10%
64	RmsProp	2000	2	94.17%	76.86%
64	RmsProp	1500	50	93.33%	70.83%
64	Adam	1000	50	85.00%	72.50%
64	RmsProp	1000	25	92.50%	72.50%
64	Adam	1000	25	95.00%	71.67%
32	Adam	2000	2	93.22%	77.69%
64	RmsProp	2000	25	85.83%	69.17%
16	SGD	1500	2	89.17%	75.83%
64	SGD	1000	25	95.00%	69.17%
16	SGD	1500	50	95.00%	79.17%
32	Adam	1500	25	95.00%	71.67%
16	SGD	2000	25	90.00%	48.76%
16	RmsProp	1000	2	95.00%	67.50%
16	Adam	2000	50	92.50%	37.19%
32	Adam	1500	2	95.83%	80.83%
64	RmsProp	1000	2	95.00%	72.50%
32	SGD	1500	25	89.17%	78.33%
32	SGD	2000	2	93.33%	65.29%
64	Adam	1500	2	96.67%	89.15%
32	SGD	1000	25	86.67%	70.00%
16	Adam	2000	25	92.50%	78.51%

Resultados de la prueba 3 (Tabla 20-Tabla 22).

Ultima prueba realizada de la repetición de pruebas con las mismas combinaciones.

Tabla 20. Resultados de la prueba 3 del diseño factorial 3<sup>4</sup>

Prueba 3					
Numero de capas ocultas	Optimizador	Iluminación	Componentes	Cámara	
				1.3M	8M
				Exactitud	Exactitud
16	SGD	1500	25	93.33%	75.00%
64	RmsProp	1500	2	97.50%	78.33%
16	RmsProp	1500	25	94.17%	76.67%
16	RmsProp	1000	50	85.00%	68.33%
16	SGD	2000	2	92.50%	78.51%
16	SGD	1000	50	95.83%	70.83%
32	RmsProp	1500	50	89.17%	73.33%
32	SGD	1000	2	95.83%	68.33%
64	SGD	2000	50	90.00%	76.86%
16	SGD	2000	50	94.17%	56.20%
32	Adam	1000	50	95.00%	68.33%
32	RmsProp	1000	25	91.67%	68.33%
16	Adam	1500	50	89.17%	80.00%
16	RmsProp	1000	25	94.17%	64.17%
16	Adam	1500	2	95.83%	75.83%
32	Adam	1000	25	95.83%	70.83%
16	Adam	1000	50	87.50%	64.17%
32	SGD	2000	25	93.33%	75.21%
64	SGD	1500	2	97.50%	77.50%
32	RmsProp	1000	50	95.83%	70.00%
16	SGD	1000	25	95.00%	68.33%
16	RmsProp	1500	50	98.33%	79.17%
32	Adam	2000	25	93.33%	49.59%
32	Adam	1000	2	83.33%	69.17%
64	Adam	2000	2	92.50%	50.41%
16	Adam	2000	2	95.00%	65.29%
64	Adam	2000	50	94.17%	65.29%
64	RmsProp	1000	50	94.17%	70.00%
32	SGD	2000	50	92.50%	49.59%
32	RmsProp	2000	25	93.33%	78.51%
64	SGD	2000	2	78.33%	76.86%
32	Adam	1500	50	80.00%	79.17%

Tabla 21. Resultados de la prueba 3 del diseño factorial 3<sup>4</sup> (2)

64	Adam	1500	50	87.50%	78.33%
32	RmsProp	2000	2	91.67%	76.67%
32	RmsProp	1000	2	94.17%	75.83%
16	Adam	1000	2	95.00%	73.33%
32	RmsProp	2000	50	89.17%	78.51%
32	SGD	1500	50	82.50%	76.67%
32	Adam	2000	50	91.67%	49.59%
16	Adam	1500	25	95.00%	80.00%
16	Adam	1000	25	95.83%	70.83%
32	RmsProp	1500	25	88.33%	75.83%
32	SGD	1000	50	85.83%	69.17%
16	RmsProp	2000	50	94.17%	77.50%
64	Adam	1000	2	95.83%	75.00%
64	SGD	1500	25	95.83%	75.83%
32	RmsProp	1500	2	97.50%	76.67%
16	RmsProp	2000	25	96.67%	75.00%
64	RmsProp	2000	50	96.67%	77.69%
64	Adam	2000	25	91.67%	49.59%
16	SGD	1000	2	92.50%	70.83%
16	RmsProp	2000	2	91.67%	75.21%
64	SGD	1500	50	80.00%	73.33%
64	SGD	1000	2	95.00%	69.17%
64	RmsProp	1500	25	97.50%	76.67%
64	SGD	2000	25	95.00%	88.43%
64	SGD	1000	50	94.17%	66.67%
64	Adam	1500	25	94.17%	80.00%
32	SGD	1500	2	94.17%	78.33%
16	RmsProp	1500	2	90.83%	85.83%
64	RmsProp	2000	2	91.67%	82.64%
64	RmsProp	1500	50	93.33%	70.83%
64	Adam	1000	50	85.00%	70.00%
64	RmsProp	1000	25	87.50%	65.00%
64	Adam	1000	25	95.00%	74.17%
32	Adam	2000	2	95.87%	79.25%
64	RmsProp	2000	25	94.17%	68.33%
16	SGD	1500	2	96.67%	77.50%
64	SGD	1000	25	95.00%	70.83%
16	SGD	1500	50	96.67%	80.00%
32	Adam	1500	25	97.50%	70.83%

Tabla 22. Resultados de la prueba 3 del diseño factorial  $3^4$  (3)

16	SGD	2000	25	94.17%	57.85%
16	RmsProp	1000	2	87.50%	76.67%
16	Adam	2000	50	94.17%	52.89%
32	Adam	1500	2	89.17%	81.67%
64	RmsProp	1000	2	93.33%	77.50%
32	SGD	1500	25	89.17%	80.00%
32	SGD	2000	2	95.00%	75.21%
64	Adam	1500	2	93.33%	78.33%
32	SGD	1000	25	95.83%	75.83%
16	Adam	2000	25	94.17%	82.64%

Las tablas también se encuentran en el repositorio de GitHub, en el siguiente link <https://github.com/MayraMateo0/Clasificacion-de-piezas-met-licas-CNN-PCA>. Además, se incluyen: el dataset utilizado, código fuente y los formularios utilizados.

Se incluye también la hoja de datos del luxómetro UT382 Series en la Tabla 23 (Ingprosuppliers, 2021):

*Tabla 23. Hoja de datos de luxómetro UT382 Series.*

<b>Función</b>	<b>Rango de resolución</b>	<b>Precisión</b>	<b>Tolerancia <math>\pm</math> ( a % lectura + b dígitos)</b>
Iluminancia medición (LUX)	20Lux	0,01Lux	$\pm$ (3% + 20)
	200Lux	0,1Lux	$\pm$ (3% + 8)
	2000Lux	1Lux	$\pm$ (3% + 8)
	20000Lux	10Lux	$\pm$ (3% + 8)
Iluminancia medición (FC)	2FC	0.001FC	La tolerancia de precisión de la FC puede verificarse mediante la conversión de unidades: FC=10,76lux, si se necesita verificar la tolerancia de precisión, se puede lograr por medio de la conversión de unidades.
	20FC	0,01FC	
	200FC	0,1FC	
	2000FC	1FC	

Hoja de datos de la cámara de 1.3 MP (European machine vision association, 2006):

<b>Cámara</b>	<b>Basler acA1300-200um</b>
Resolución (H x V pixeles)	1280 x 1024
Sensor	EV76C661
Tamaño del sensor (óptico)	1/1.8"
Tecnología de sensores	CMOS,global shutter
Tamaño del pixel ( $\mu\text{m}$ )	5.3 x 5.3
Velocidad de fotogramas	60
Mono / Color	Mono
Formato de salida de vídeo	Mono 8, Mono 12, Mono 12 p, YUV 4:2:2 p, YUV 4:2:2 (YUYV) p, Bayer BG 8, Bayer BG 12, Bayer BG 12 p: Bayer GB (Aptina), Bayer GR (CMOSIS)
Interfaz	Fast Ethernet (100 Mbit/s) or Gigabit Ethernet (1000 Mbit/s)
Sincronización	A través de un disparador externo, a través de la conexión Ethernet o de forma libre
Control de la exposición	Mediante un disparador externo o programable a través de la API de la cámara.

## ACRONIMOS

**HOG:** Histogram of Oriented Gradients.

**HAAR:** Clasificador para la detección de objetos.

**SIFT:** Scale-Invariant Feature Transform

**PCA:** Principal Component Analysis

**FREAK:** Fast Retina Keypoint

**DPI:** Dots Per Inch

**CMOS:** Complementary Meta Oxide Semiconductor Memory