


# CONESCAPANHONDURAS2025paper20.pdf

 Institute of Electrical and Electronics Engineers (IEEE)

---

## Document Details

### Submission ID

trn:oid:::14348:477751470

### Submission Date

Jul 31, 2025, 11:54 PM CST

### Download Date

Aug 12, 2025, 12:19 PM CST

### File Name

CONESCAPANHONDURAS2025paper20.pdf

### File Size

355.9 KB

6 Pages




4,456 Words

25,837 Characters

# 15% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

## Top Sources

- 14%  Internet sources
- 11%  Publications
- 0%  Submitted works (Student Papers)

## Integrity Flags




### 0 Integrity Flags for Review

No suspicious text manipulations found.

Our system's algorithms look deeply at a document for any inconsistencies that would set it apart from a normal submission. If we notice something strange, we flag it for you to review.

A Flag is not necessarily an indicator of a problem. However, we'd recommend you focus your attention there for further review.

## Top Sources

- 14%  Internet sources
- 11%  Publications
- 0%  Submitted works (Student Papers)

## Top Sources

The sources with the highest number of matches within the submission. Overlapping sources will not be displayed.

|    |             |  |     |
|----|-------------|--|-----|
| 1  | Internet    | arxiv.org  | 1%  |
| 2  | Internet    | upcommons.upc.edu  | 1%  |
| 3  | Internet    | ouci.dntb.gov.ua   | 1%  |
| 4  | Publication | R. Manjula Devi, Rajesh Kumar Dhanaraj, Subhendu Kumar Pani, Rashmi Prava D...       | <1% |
| 5  | Internet    | www.medrxiv.org  | <1% |
| 6  | Internet    | edoc.ub.uni-muenchen.de  | <1% |
| 7  | Internet    | kuet.ac.bd   | <1% |
| 8  | Internet    | www.researchgate.net   | <1% |
| 9  | Publication | Nalla Praveen, Narinder Singh Pun, Sanjay Kumar Sonbhadra, Sonali Agarwal, M...      | <1% |
| 10 | Publication | Sanjeev Gupta, Ashish Mishra. "Deep transfer learning-based classification of Whi... | <1% |
| 11 | Publication | "Human-Computer Interaction", Springer Science and Business Media LLC, 2025          | <1% |

|    |             |  |     |
|----|-------------|--|-----|
| 12 | Publication | Bana Fridath Bio Nigan, Alban Gildas Zohoun, Ahmed Dooguy Kora. "White Blood ... | <1% |
| 13 | Internet    | www.coursehero.com   | <1% |
| 14 | Internet    | ejournal.uin-suka.ac.id  | <1% |
| 15 | Internet    | export.arxiv.org   | <1% |
| 16 | Internet    | reunir.unir.net  | <1% |
| 17 | Internet    | ebuah.uah.es   | <1% |
| 18 | Internet    | constellation.uqac.ca  | <1% |
| 19 | Internet    | journal.ittelkom-pwt.ac.id   | <1% |
| 20 | Internet    | link.springer.com  | <1% |
| 21 | Internet    | scholars.uthscsa.edu   | <1% |
| 22 | Internet    | www.drgdiaz.com  | <1% |
| 23 | Internet    | www.symphonyai.com   | <1% |
| 24 | Internet    | galladorafael.github.io  | <1% |
| 25 | Internet    | sciendo.com  | <1% |

|    |             |  |     |
|----|-------------|--|-----|
| 26 | Internet    | sedici.unlp.edu.ar   | <1% |
| 27 | Internet    | ejournal.uhb.ac.id   | <1% |
| 28 | Internet    | mediatum.ub.tum.de   | <1% |
| 29 | Internet    | prezi.com  | <1% |
| 30 | Internet    | revistasinvestigacion.unmsm.edu.pe   | <1% |
| 31 | Internet    | www.cs.mta.ac.il   | <1% |
| 32 | Internet    | www.icsh.org   | <1% |
| 33 | Internet    | www.scielo.br  | <1% |
| 34 | Publication | Thangaprakash Sengodan, Sanjay Misra, M Murugappan. "Advances in Electrical ...      | <1% |
| 35 | Publication | V. Sharmila, S. Kannadhasan, A. Rajiv Kannan, P. Sivakumar, V. Vennila. "Challeng... | <1% |

# Modelo de Clasificación de Leucocitos Individuales Basado en Inteligencia Artificial

**Abstract**—The manual classification of leukocytes is the first step in blood smear analysis. However, as it is a process performed under microscope observation, it is time-consuming and prone to human errors. Addressing this issue, a method based on artificial intelligence was developed using transfer learning. For its implementation, a literature review was conducted to select an appropriate approach for image processing, identify public datasets with the necessary quality and volume, and establish metrics to evaluate the model's performance at both general and specific levels. The project successfully achieved the goal of developing an automated method for classifying individual leukocytes, with the potential to support clinical diagnosis. Based on transfer learning, the model was evaluated using performance metrics, demonstrating high efficiency in identifying complex patterns in microscopic images, even in the presence of cellular diversity. Its ability to accurately classify both mature and immature leukocytes, which are clinically relevant, highlights its potential as a promising tool for optimizing diagnostic processes, where precision is critical. This solution could reduce analysis times, minimize human errors, and serve as a valuable complement in detecting pathologies associated with leukocyte abnormalities.

**Index Terms**—leukocytes, artificial intelligence, classification, individual cells.

## I. INTRODUCCIÓN

En hematología, afecciones como la Leucemia Mieloide Aguda alteran el ciclo de vida celular de los leucocitos, impidiendo su maduración y causando una multiplicación no funcional de estos. Según C. Récher y F. Huguet [1], “este cáncer representa el 80 % de las leucemias agudas en adultos, siendo más común en hombres, y su incidencia aumenta con la edad, especialmente después de los 50 años”. En este contexto médico, el personal de laboratorio clínico es clave para evaluar diagnósticos mediante muestras de sangre y biopsias de médula ósea. Aunque la clasificación microscópica de células en frotis sanguíneos es fundamental, este método manual puede generar errores. Por ello, herramientas como la inteligencia artificial resultan esenciales: permiten procesar grandes volúmenes de datos y mejorar la precisión y eficiencia de los diagnósticos.

En este proyecto se desarrolló un algoritmo de inteligencia artificial para clasificar leucocitos individuales, capaz de identificar distintos tipos celulares. Esta herramienta se diseñó para apoyar el análisis de imágenes de frotis sanguíneos, reduciendo errores diagnósticos asociados a la clasificación manual en la práctica clínica.

## II. OBJETIVO

Desarrollar un método de clasificación de leucocitos individuales por medio de inteligencia artificial utilizando Python

como lenguaje de programación, tal que sea de apoyo para el personal clínico de laboratorio en la identificación de estas células en frotis de sangre.

## III. ESTADO DEL ARTE

### A. Hematología

En este ámbito se implementan técnicas de aprendizaje supervisado a partir de conjuntos de datos clasificados por características específicas para entrenar algoritmos en el reconocimiento de células. Estos datos se organizan en base a la morfología celular para diagnosticar enfermedades caracterizadas por malformaciones celulares [2]. Como ejemplo de aplicación en este campo, miembros de la Universidad de Judendo, en asociación con la Corporación Sysmex en Tokio, Japón, diseñaron un algoritmo basado en redes convolucionales para un sistema de apoyo a la decisión clínica para el diagnóstico de un tipo de anemia utilizando frotis de sangre e imágenes de la muestra para reconocer estas células características de la enfermedad [3].

### B. Clasificación de células sanguíneas

La identificación de células de la sangre, siendo fundamental para el diagnóstico de enfermedades, puede verse afectada por la baja resolución de las imágenes obtenidas mediante microscopios y los errores en las tinciones. En este contexto, [4] propusieron un enfoque basado en inteligencia artificial para mejorar la calidad y precisión en la clasificación de imágenes de leucocitos, empleando técnicas avanzadas de procesamiento. De forma similar, la integración de herramientas, como la visión computacional, ha demostrado ser efectiva para optimizar tanto la segmentación como la clasificación celular. Sin embargo, aunque [5] han comprobado que la combinación de visión computacional e IA mejora la identificación celular, persisten limitaciones debido a la escasa variedad de muestras utilizadas en los entrenamientos. Esto sugiere que futuras investigaciones deberían enfocarse en optimizar la adquisición de imágenes para aumentar la fiabilidad de los modelos.

### C. Aprendizaje por transferencia

Este tipo de aprendizaje utiliza modelos preentrenados para tareas relacionadas, mejorando la extracción de características en imágenes médicas [6]. Es especialmente útil con conjuntos de datos pequeños, donde estos modelos, muchos con precisiones superiores al 90 %, permiten clasificar eficientemente glóbulos blancos, bacterias y otros elementos sanguíneos [7]. Aunque técnicas como aumento de datos y segmentación pueden optimizar los resultados, el rendimiento final depende tanto del modelo base seleccionado como de las estrategias complementarias aplicadas.

## IV. METODOLOGÍA

### A. Enfoque

Este proyecto se desarrolló bajo un enfoque mixto, ya que combina el análisis de datos numéricos obtenidos mediante métricas y gráficos aplicados a la arquitectura del algoritmo con las características visuales de las imágenes utilizadas para el entrenamiento del modelo. Las métricas permiten medir la precisión del modelo en la clasificación de imágenes, facilitando una evaluación cuantitativa de su rendimiento. Al mismo tiempo, la consideración de las características visuales asegura que el modelo se base en información clave de éstas para la tarea, aumentando su capacidad de generalización. Además, el proyecto adopta un alcance descriptivo centrado en detallar las propiedades y características de la arquitectura del algoritmo diseñado, y presentar los resultados obtenidos a partir de las métricas implementadas.

### B. Variables de Investigación

Las variables de la investigación se clasificaron en independientes y dependientes. La variable independiente se enfocó en la arquitectura del modelo de clasificación, mientras que las variables dependientes incluyeron el modelo resultante, el tiempo de entrenamiento y el rendimiento del algoritmo.

### C. Técnicas e instrumentos aplicados

- Revisión bibliográfica. Realizada para elegir el modelo del algoritmo de clasificación, seleccionar métricas de evaluación y definir los conjuntos de datos de entrenamiento. También proporcionó guías para la programación del modelo.
- Métricas. Aplicadas para evaluar el rendimiento del algoritmo mediante valores numéricos y gráficos que reflejan la asertividad del modelo en la clasificación de imágenes.

### D. Materiales

- Conjunto de imágenes de tipos de leucocitos maduros e inmaduros clasificados, obtenidos de The Cancer Image Archive (TCIA) y Mendeley Data.
- Jupyter Notebooks como entorno de programación.
- Librerías para procesamiento de imágenes y manejo de datos, desarrollo del algoritmo, cálculos matemáticos, generación de gráficos y optimización.
- Computadora portátil con 16 GB de RAM.
- Guías técnicas de apoyo conceptual y para el desarrollo de bloques de programación.

### E. Metodología de Estudio

La revisión bibliográfica permitió definir el modelo de IA a desarrollar, analizando enfoques recientes para clasificación celular; así como la selección de dos conjuntos de datos empleados en diversos estudios científicos sobre clasificación de células sanguíneas.

En cuanto a la redacción de la estructura del código, la principal herramienta guía fue el libro de "Chollet, F. (2017). *Deep learning with python*. Manning Publications", de donde se obtuvieron las ideas principales para la construcción de los bloques del algoritmo. Como guía complementaria, se utilizó el libro de "Zhang, A., Lipton, Z. C., Li, M., & Smola, A. J. (2021). *Dive into Deep Learning*".

Por parte de la búsqueda literaria, fueron establecidas las librerías de TensorFlow y Keras para la creación del modelo

de clasificación. Scikit-Learn aplicada para evaluar métricas y división del conjunto de datos. Para el procesamiento de imágenes, se definió CV2, mientras que NumPy para cálculos matemáticos. El análisis de datos y diseño de gráficos por medio de Pandas, Seaborn y Matplotlib.

### F. Metodología de Validación

Se realizó una comparación de los resultados obtenidos con las métricas reportadas en otros estudios que implementaron aprendizaje profundo por transferencia en la clasificación de leucocitos. Estos estudios se muestran en la Tabla 2 presente en la sección de Resultados.

## V. RESULTADOS

El desarrollo del algoritmo de clasificación de imágenes siguió tres etapas esenciales: la preparación del conjunto de datos para el entrenamiento, el desarrollo del modelo de procesamiento y clasificación de imágenes, y la evaluación del rendimiento mediante diversas métricas. A continuación, se detalla la aplicación de estos procedimientos en el desarrollo del modelo de clasificación de leucocitos.

### A. Selección de Conjunto de Datos

El primer conjunto de datos seleccionado, propuesto por [8], contiene 14 clases de leucocitos con imágenes en formato TIFF de  $400 \times 400$  píxeles. Este conjunto está desbalanceado, ya que la cantidad de imágenes por clase varía considerablemente. En cambio, el segundo conjunto de [9] incluye 8 clases de leucocitos, con imágenes en formato JPG y un tamaño de  $360 \times 363$  píxeles. De estas 8 clases, 7 están claramente diferenciadas, mientras que la octava agrupa 4 clases de leucocitos inmaduros. Ambos conjuntos contienen imágenes etiquetadas, lo que facilita su identificación y clasificación.

Las imágenes difieren, además, en su método de adquisición: el primer conjunto fue capturado mediante microscopía electrónica, mientras que el segundo se obtuvo con un digitalizador de portaobjetos. También presentan variaciones en tonalidad, posiblemente debido a diferencias en el método de tinción, el procesamiento post-captura y características de los equipos de digitalización (balance de luz, calidad de cámara y lentes).

Considerando la cantidad de imágenes, sus diferencias en formato y la importancia clínica de las clases presentes, se conformó un nuevo conjunto balanceado con 10 clases de leucocitos, tanto maduros como inmaduros, relevantes en el estudio de patologías sanguíneas y el funcionamiento del sistema inmunológico. Para garantizar una adecuada generalización del algoritmo, se priorizaron las clases con suficiente representación.

Luego de la selección, se normalizó el formato de todas las imágenes a JPG y se redimensionaron a  $400 \times 400$  píxeles; algunas imágenes presentaron pérdida de píxeles, lo que resultó en la aparición de una banda negra, como se muestra en el ejemplo de la ilustración 1. Estas imágenes fueron eliminadas para evitar introducir ruido al modelo.

Tras esta eliminación, el conjunto de entrenamiento conformado contuvo un total de 12,450 imágenes. La Tabla 1

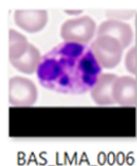


Ilustración 1. Imagen con pérdida de información por conversión de formato.

muestra la cantidad de imágenes utilizadas por cada clase de leucocitos para entrenar el algoritmo.

TABLA I  
DISTRIBUCIÓN DE IMÁGENES POR CLASE DE LEUCOCITOS

| Clase de leucocito    | Tipo     | Cantidad total |
|-----------------------|----------|----------------|
| Basófilo              | Maduro   | 1,295          |
| Eosinófilo            |          | 1,321          |
| Linfocito             |          | 1,297          |
| Monocito              |          | 1,300          |
| Neutrófilo segmentado |          | 1,294          |
| Neutrófilo en banda   | Inmaduro | 1,308          |
| Mieloblasto           |          | 1,764          |
| Mielocito             |          | 1,179          |
| Promielocito          |          | 662            |
| Metamielocito         |          | 1,030          |

### B. Construcción del Modelo

La programación fue realizada en Jupyter Notebooks siguiendo un desarrollo secuencial de 13 partes que conformaron el modelo diseñado.

El diseño del algoritmo comenzó con la importación de las librerías necesarias para el manejo de datos, procesamiento de las imágenes, división del conjunto de datos, diseño de la arquitectura y aplicación de métricas.

A través de la aplicación de un dataframe, se asignó el nombre de la clase correspondiente a cada una de las imágenes para trabajar bajo aprendizaje supervisado.

Se redimensionaron las imágenes a un tamaño de 128 x 128 píxeles en un canal de 3 colores (RGB) ya que son imágenes a color, facilitando así el procesamiento de los datos por parte del algoritmo. El conjunto se dividió, considerando consejos generales de las guías técnicas, en un 80 % para la fase de entrenamiento, 12 % para validación y 8 % para prueba, con la definición del tamaño de lote de imágenes a ser procesadas por iteración durante la fase de entrenamiento, con la aplicación de una semilla (seed) para asegurar la reproducibilidad de los resultados.

Para la arquitectura, se definió el aprendizaje profundo por transferencia con ResNet50 como modelo preentrenado para extraer información, sin pérdida de esta, entre las capas del modelo. Este aprendizaje aprovechó el conocimiento adquirido por este modelo, el cual cuenta con 50 capas de procesamiento y utiliza bloques residuales que facilitan la transmisión eficiente de información y ayudan a evitar problemas como

el desvanecimiento del gradiente [10]. Para esta arquitectura, se emplearon únicamente las capas inferiores, encargadas de captar características generales aprendidas en su entrenamiento con un conjunto de datos amplio, las cuales se procesan por el resto de las capas de la arquitectura.

La arquitectura constó de las siguientes capas:

- Modelo preentrenado: detección y extracción de patrones básicos como de las imágenes.
- Capa de normalización: para estabilizar el entrenamiento y la activación entre capas, evitando la pérdida de información relevante. Posee dos parámetros que suavizan los cambios en las mediciones del modelo y evitan errores en los cálculos, respectivamente.
- Capa densa: con una función de activación ReLU y 256 neuronas, se detectan los patrones complejos de las imágenes y se procesan aquellos extraídos por el modelo para aprender de ellos. Esta función ReLU fue seleccionada por su eficiencia en el aprendizaje de características complejas y su capacidad para evitar la pérdida de información en redes multicapa. Además, se añadió un regularizador para penalizar pesos grandes, evitando que el algoritmo dependa de características específicas.
- Dropout: para desactivar un porcentaje aleatorio de neuronas durante el entrenamiento, lo que previene que el modelo memorice patrones específicos y mejora su capacidad de generalización.
- Salida: empleó una función de activación Softmax, que convierte las salidas en probabilidades por clase, permitiendo al modelo predecir la categoría más probable para cada imagen.

Para el entrenamiento, se utilizó un optimizador para ajustar los pesos del modelo y equilibrar la velocidad de aprendizaje. Además, se aplicó una función de pérdida que mide la diferencia entre las probabilidades predichas y las etiquetas reales, ajustando los pesos para minimizar el error. Finalmente, el establecimiento de una tasa de aprendizaje para regular la actualización de los pesos durante el entrenamiento.

Para monitorear la calidad de los resultados, se aplicó una detención temprana respecto a la precisión de validación para un entrenamiento de 10 épocas; es decir, que si este valor no mejoraba en 5 épocas consecutivas, el entrenamiento se detendría.

Con un entrenamiento final de 10 épocas, cada una de estas tardó aproximadamente 25 minutos en procesarse, siendo un total aproximado de 4 horas, y por cada época se procesaron alrededor de 312 iteraciones. El tiempo dependió de la cantidad de épocas configuradas para el entrenamiento; la tasa de aprendizaje, el tamaño de las imágenes a ser procesadas y los parámetros configurados en las capas de la arquitectura.

### C. Métricas

- Las gráficas de evolución de precisión y pérdida (Ilustración 2) permitieron evaluar el rendimiento del algoritmo en entrenamiento y validación. La precisión de entrenamiento (gráfica azul) mostró un aumento progresivo, indicando que el modelo aprendió de los datos. En cambio, la precisión de validación (gráfica naranja) fluctuó entre 0.80 y 0.89,



posiblemente debido a la complejidad de las imágenes, ruido, pérdida de calidad por conversión de formato o similitudes entre clases; esta fluctuación se observa con la variación de picos en este gráfico. Aun así, su estabilidad en este rango sugiere que el modelo logró identificar patrones útiles para la clasificación.

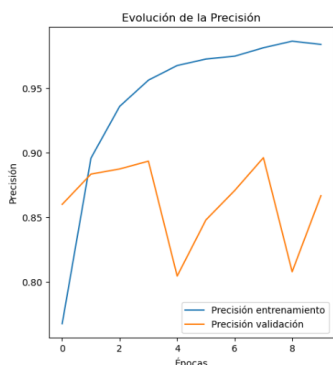


Ilustración 2. Evolución de precisión.

La reducción constante de la pérdida, observada en la gráfica de la ilustración 3 en el conjunto de validación, indica que el modelo estaba aprendiendo eficazmente de los datos de entrenamiento y generalizando adecuadamente a los datos de validación. A lo largo de las épocas, la pérdida mantuvo una tendencia de descenso constante.

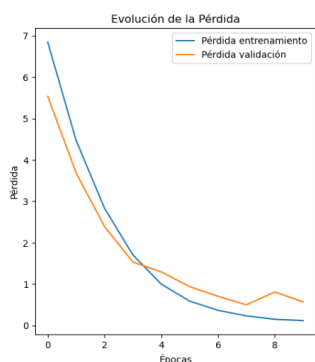


Ilustración 3. Evolución de pérdida.

- Se aplicó un historial de precisión y pérdida para las 3 fases del algoritmo.

Durante la fase de entrenamiento, el modelo alcanzó una precisión del 99.17 %, lo que indica que aprendió correctamente de la gran mayoría de las muestras. La pérdida obtenida fue de 0.1670, sugiriendo que la función de pérdida se optimizó de manera adecuada.

En la fase de validación, el modelo logró una precisión del 89.62 %, lo que sugiere una buena capacidad de generalización a datos no vistos durante el entrenamiento.

Para la fase de prueba, la precisión fue del 90.27 %, lo que confirma su capacidad para clasificar correctamente datos completamente nuevos. La pérdida en este conjunto fue de 0.4685, indicando que el margen de error se mantuvo similar al

de la validación. Esto sugiere que no hubo un sobreajuste significativo y que el modelo es capaz de clasificar correctamente la mayoría de las imágenes, aunque presentó dificultades con algunas muestras debido, posiblemente, a los factores de las imágenes previamente mencionados.

- El reporte de clasificación, presente en la Ilustración 4, se aplicó a la fase de prueba con 997 imágenes. Este incluye las métricas de precisión, sensibilidad y puntuación F1 para evaluar el rendimiento del modelo en cada clase, además de la métrica support, que indica la cantidad de imágenes utilizadas por clase.

Las clases basófilo, eosinófilo, linfocito, mieloblasto y monocito lograron una puntuación F1 superior a 0.98 y una precisión de 0.95, reflejando un aprendizaje eficaz. En contraste, los metamielocitos y neutrófilos en banda tuvieron una precisión menor (0.84 y 0.82, respectivamente), aunque con un rendimiento aceptable. La clase neutrófilo segmentado mostró una sensibilidad de 0.84, mientras que los neutrófilos en banda alcanzaron 0.82, evidenciando cierta dificultad en su clasificación.

Las clases con menor desempeño fueron mielocito y promielocito, con una puntuación F1 de 0.78 y 0.68, respectivamente, y una precisión de 0.61 en promielocitos.

| Reporte de Clasificación: |           |        |          |         |
|---------------------------|-----------|--------|----------|---------|
|                           | precision | recall | f1-score | support |
| Basófilo                  | 0.97      | 0.98   | 0.97     | 122     |
| Eosinófilo                | 0.99      | 0.99   | 0.99     | 110     |
| Linfocito                 | 0.98      | 0.98   | 0.98     | 102     |
| Metamielocito             | 0.84      | 0.85   | 0.84     | 79      |
| Mieloblasto               | 0.95      | 1.00   | 0.98     | 146     |
| Mielocito                 | 0.82      | 0.75   | 0.78     | 91      |
| Monocito                  | 1.00      | 0.89   | 0.94     | 109     |
| Neutrófilo_Segmentado     | 0.85      | 0.84   | 0.84     | 98      |
| Neutrófilo_en_Banda       | 0.82      | 0.82   | 0.82     | 93      |
| Promielocito              | 0.61      | 0.77   | 0.68     | 47      |
| accuracy                  |           |        | 0.90     | 997     |
| macro avg                 | 0.88      | 0.89   | 0.88     | 997     |

Ilustración 4. Reporte de Clasificación.

- La matriz de confusión (Ilustración 5) muestra las clasificaciones correctas y erróneas en la fase de prueba por cada clase respecto a las otras. La clase con mejor desempeño fue Mieloblasto, con 146 aciertos y ningún error, logrando una precisión casi perfecta.

En contraste, se observaron dificultades en la clasificación de leucocitos inmaduros y neutrófilos. Por ejemplo, Mielocito tuvo 68 aciertos y 23 errores. La confusión que se destaca fue entre Neutrófilo en Banda y Neutrófilo Segmentado, lo que sugiere similitudes morfológicas. De manera similar, Promielocito obtuvo 36 aciertos y 11 errores, siendo confundido principalmente con Mielocito.

- Por último, la Curva ROC fue útil para evaluar el rendimiento del modelo en la clasificación multiclase. Esta curva, representada en distintos colores para cada clase, muestra la confianza del modelo en sus predicciones al transformar los valores en un rango de 0 a 1. Un valor inferior a 0.5, ubicado por debajo de la línea punteada de la gráfica, indica una clasificación incorrecta. Este gráfico brinda confianza al

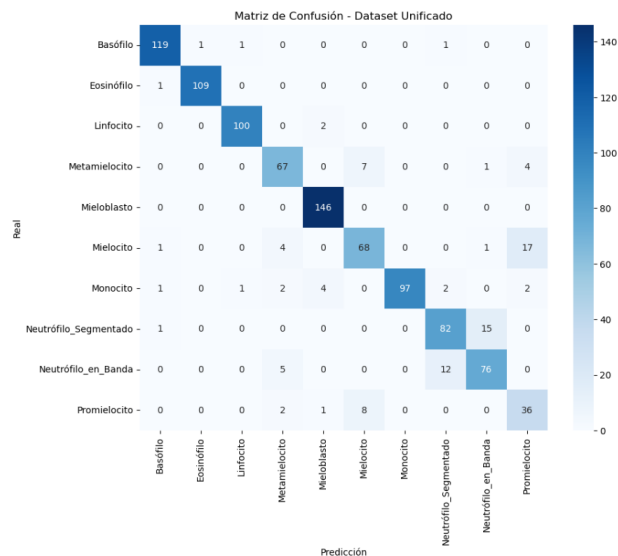


Ilustración 5. Matriz de confusión.

personal clínico respecto a la alta precisión que puede tener el algoritmo respecto a la clasificación de leucocitos que es capaz de identificar.

En la Ilustración 6, se observa que la mayoría de las clases como Basófilo, Eosinófilo, Linfocito, Mieloblasto y Monocito lograron un AUC de 1.00, lo que indica una clasificación de alta confiabilidad. Otras clases, como Metamielocitos y Neutrófilos, obtuvieron un AUC de 0.99, mostrando un rendimiento alto con una ligera posibilidad de confusión, pero aún así brindando resultados confiables.

Las clases con menor desempeño fueron Mielocito (0.98) y Promielocito (0.97); sin embargo, aunque sus valores siguen siendo elevados, indicando un buen rendimiento general del modelo, sigue presente la posibilidad, aunque en menor probabilidad, de que estas clases de leucocitos inmaduros sean clasificados erróneamente.

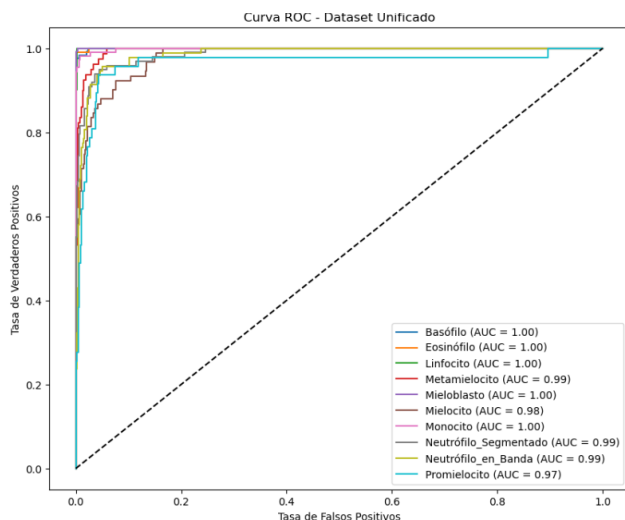


Ilustración 6. Curva ROC.

## D. Validación de resultados

Para validar el resultado del modelo, se comparó el porcentaje de precisión obtenido en la fase de prueba con los valores reportados en los estudios de la Tabla 2, cuyos modelos fueron entrenados con cinco clases de leucocitos (basófilos, eosinófilos, linfocitos, monocitos y neutrófilos), utilizando también ResNet50 como modelo preentrenado. Estos estudios emplearon leucocitos maduros debido a su estructura definida, lo que facilita su identificación, a diferencia de los leucocitos inmaduros.

El modelo desarrollado alcanzó un 90.2 % de precisión en diez clases, superando el rendimiento de estudios previos que trabajaban con la mitad de categorías, únicamente con los de tipo maduro. Este resultado confirma su capacidad para generalizar en problemas complejos y brinda confianza en su aplicabilidad clínica en la clasificación extendida de leucocitos, obteniendo así mejores resultados en esta labor.

TABLA II  
PORCENTAJES DE PRECISIÓN DE MODELOS DE CLASIFICACIÓN CON RESNET50

| Estudio realizado  | Precisión |
|--|-----------|
| White blood cells classification using multi-fold pre-processing and optimized CNN model [12].                                   | 74.58 %   |
| White Blood Cell Classification: A Comparison between VGG-16 and ResNet-50 Models [13].  | 88.29 %   |
| Deep transfer learning-based classification of White Blood Cells using customized classification base [14].                      | 98 %      |
| Improved Classification of White Blood Cells with the Generative Adversarial Network and Deep Convolutional Neural Network [15]. | 93.3 %    |

## VI. DISCUSIÓN DE RESULTADOS

El modelo de clasificación diseñado alcanzó una precisión general del 90.27 % en la fase de prueba, un resultado destacable considerando la amplia variedad de tipos de leucocitos analizados en comparación con otros modelos diseñados, como aquellos con los que se validó este resultado previamente. Esta alta precisión refleja el potencial del algoritmo para optimizar la clasificación manual de leucocitos, mejorando la calidad del diagnóstico al identificar tanto células maduras como inmaduras clínicamente relevantes en el análisis clínico de frotis a través de imágenes microscópicas.

Otra gran ventaja de este modelo de clasificación radica en su adaptación a variaciones en las características visuales de las imágenes de las células, gracias a su entrenamiento con dos conjuntos de datos independientes. Cada conjunto presenta imágenes con diferencias atribuibles a los métodos de digitalización y obtención multimedia, permitiendo que el modelo desarrollara robustez frente a distorsiones, artefactos o cambios en iluminación, resolución y formato, mejorando así su precisión para identificar células en diferentes condiciones.

Además, al aprender patrones generalizables a partir de dicha diversidad, el algoritmo evitó significativamente el sobreajuste, haciéndolo flexible para aplicaciones en entornos clínicos o de investigación con variabilidad en los datos de entrada.

Una limitación fue el desequilibrio en la cantidad de imágenes en el conjunto de datos, lo que afectó el rendimiento en clases minoritarias como promielocitos y mielocitos, evidenciado por sus bajos puntajes F1. Esto sugiere que el modelo tuvo dificultades para aprender patrones en categorías poco representadas, un desafío común en modelos de clasificación. Aunque la precisión global fue alta, este valor puede ser engañoso en contextos clínicos, ya que un modelo con sesgo hacia clases mayoritarias podría generar falsas clasificaciones en categorías con menos muestras representativas, comprometiendo el diagnóstico. Por ello, se evaluó con una validación cruzada por medio del análisis de diversas métricas para una evaluación más integral y detallada del desempeño por clase.

Se observó una brecha moderada entre la pérdida de entrenamiento (0.1671) y validación (0.5012), lo que indica cierto grado de sobreajuste o dificultad para generalizar en imágenes no vistas. Esta discrepancia podría mitigarse mediante técnicas de regularización, ajustes arquitectónicos o un conjunto de datos más equilibrado y homogéneo en formato. Los resultados de la curva ROC respaldan la fiabilidad del modelo en entornos clínicos. Todas las clases mostraron valores AUC superiores a 0.97, lo que demuestra una alta capacidad para distinguir entre diversos tipos de leucocitos. Esta alta discriminación es crucial para garantizar decisiones clínicas precisas, minimizando errores que podrían derivar en diagnósticos incorrectos o tratamientos inadecuados.

## VII. CONCLUSIONES

El conjunto de datos construido con 12,450 imágenes con 10 clases de leucocitos, tomó como base fundamental el uso de dos conjuntos de acceso libre en internet. Este nuevo conjunto fue diseñado para garantizar la diversidad y representatividad de las clases clave en el diagnóstico de patologías sanguíneas. Este conjunto permitió una adecuada generalización del modelo durante el entrenamiento.

La arquitectura del modelo, basada en aprendizaje por transferencia con ResNet50 y compuesta por 5 capas, aprovechó las características generales aprendidas del modelo preentrenado, mejorando la detección de estructuras celulares y reduciendo el riesgo de sobreajuste.

Las métricas como precisión, sensibilidad, y puntuación F1 proporcionaron una evaluación detallada, con una precisión del 90.27 % en la fase de prueba, lo que demostró una buena capacidad de generalización y un rendimiento robusto, incluso ante clases complejas. La matriz de confusión y la curva ROC confirmaron la efectividad del modelo para distinguir entre clases, con un AUC de 1.00 para la mayoría de las clases.

Comparándolo con estudios previos, el modelo mostró resultados competitivos, destacando su capacidad para clasificar tanto leucocitos maduros como inmaduros, alcanzando así con éxito el objetivo de desarrollar un método de clasificación de leucocitos individuales.

## BIBLIOGRAFÍA

- [1] C. Récher y F. Huguet, «Leucemia mieloide aguda», EMC - Tratado de Medicina, vol. 28, n.o 1, pp. 1-11, 2024, doi: [https://doi.org/10.1016/S1636-5410\(23\)48735-1](https://doi.org/10.1016/S1636-5410(23)48735-1).
- [2] W. Walter et al., «Artificial intelligence in hematological diagnostics: Game changer or gadget?», Blood Reviews, vol. 58, p. 101019, 2023, doi: <https://doi.org/10.1016/j.blre.2022.101019>.
- [3] K. Kimura et al., «A novel automated image analysis system using deep convolutional neural networks can assist to differentiate MDS and AA», Scientific Reports, vol. 9, n.o 1, p. 13385, sep. 2019, doi: [10.1038/s41598-019-49942-z](https://doi.org/10.1038/s41598-019-49942-z).
- [4] J. Ferdousi, S. I. Lincoln, M. K. Alom, y M. Foysal, «A deep learning approach for white blood cells image generation and classification using SRGAN and VGG19», Telematics and Informatics Reports, vol. 16, p. 100163, 2024, doi: <https://doi.org/10.1016/j.teler.2024.100163>.
- [5] B. Leng, C. Wang, M. Leng, M. Ge, y W. Dong, «Deep learning detection network for peripheral blood leukocytes based on improved detection transformer», Biomedical Signal Processing and Control, vol. 82, p. 104518, 2023, doi: <https://doi.org/10.1016/j.bspc.2022.104518>.
- [6] H. Bichri, A. Chergui, y M. Hain, «Image Classification with Transfer Learning Using a Custom Dataset: Comparative Study», Procedia Computer Science, vol. 220, pp. 48-54, 2023, doi: <https://doi.org/10.1016/j.procs.2023.03.009>.
- [7] D. S. Ametefe et al., «Automatic classification and segmentation of blast cells using deep transfer learning and active contours», International Journal of Laboratory Hematology, vol. 46, n.o 5, pp. 837-849, 2024, doi: <https://doi.org/10.1111/ijlh.14305>.
- [8] C. Matek, S. Schwarz, C. Marr, y K. Spiekermann, «A Single-cell Morphological Dataset of Leukocytes from AML Patients and Non-malignant Controls», The Cancer Imaging Archive, 2019, doi: <https://doi.org/10.7937/tcia.2019.36f5o9ld>.
- [9] A. Acevedo, A. Merino, S. Alférez, Á. Molina, L. Boldú, y J. Rodellar, «A dataset for microscopic peripheral blood cell images for development of automatic recognition systems», Mendeley Data V1, 2020, doi: [10.17632/snkd93bnjr.1](https://doi.org/10.17632/snkd93bnjr.1).
- [10] K. He, X. Zhang, S. Ren, y J. Sun, «Deep Residual Learning for Image Recognition», en 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 770-778, doi: [10.1109/CVPR.2016.90](https://doi.org/10.1109/CVPR.2016.90).
- [11] O. Saidani et al., «White blood cells classification using multi-fold pre-processing and optimized CNN model», Scientific Reports, vol. 14, n.o 1, p. 3570, feb. 2024, doi: [10.1038/s41598-024-52880-0](https://doi.org/10.1038/s41598-024-52880-0).
- [12] S. Vatathanavaro, S. Tungjitnob, y K. Pasupa, «White Blood Cell Classification: A Comparison between VGG-16 and ResNet-50 Models», 2018. [En línea]. Disponible en: <https://api.semanticscholar.org/CorpusID:202691981>.
- [13] S. Gupta y A. Mishra, «Deep transfer learning-based classification of White Blood Cells using customized classification base», en Proceedings of the 2024 Sixteenth International Conference on Contemporary Computing, IC3-2024, New York, NY, USA: Association for Computing Machinery, 2024, pp. 585-595, doi: [10.1145/3675888.3676117](https://doi.org/10.1145/3675888.3676117).
- [14] K. Almezghwi y S. Serte, «Improved Classification of White Blood Cells with the Generative Adversarial Network and Deep Convolutional Neural Network», Computational Intelligence and Neuroscience, vol. 2020, n.o 1, p. 6490479, 2020, doi: <https://doi.org/10.1155/2020/6490479>.