

Comparación de técnicas basadas en visión computacional y *machine learning* para la detección temprana de anemia a partir del análisis de uñas

Keico Anavela Heredia-Menor

keicoheredia@gmail.com / Universidad ESAN

Wilfredo Mamani-Ticona

wmamani@esan.edu.pe / Universidad ESAN

Recepción: 1/9/2020 Aceptación: 3/10/2020

RESUMEN. En el Perú la anemia es una enfermedad que está presente en más del 40 % de la población, es común tanto en niños como en adolescentes, y predomina en mujeres gestantes y niños menores de dos años, lo que compromete seriamente su desarrollo. Para diagnosticar la anemia es necesario realizar pruebas de laboratorio mediante el análisis de la sangre, donde se determinan los niveles de hemoglobina. Sin embargo, la mayoría de los hospitales no cuentan con los equipos adecuados para realizar las pruebas, lo que ocasiona retrasos en la entrega de los diagnósticos. El objetivo de esta investigación es comparar técnicas basadas en visión computacional y *machine learning* para la detección temprana de anemia a partir del análisis de uñas, de manera que los doctores puedan utilizarlo como apoyo en la detección de la anemia para un descarte temprano. Con un diagnóstico oportuno, se evitará a los pacientes padecer las diferentes etapas de esta enfermedad y especialmente a los que se encuentran en la etapa crónica cuando las consecuencias son graves, debido a que la anemia puede indicar la presencia de otra enfermedad subyacente. Se realizaron varios experimentos y los mejores resultados fueron: *Accuracy* 0,989, *precision* 0,98, *recall* 0,98 y *F1-score* de 0,98, con la arquitectura VGG19 como extractor de características en combinación con el clasificador *support vector machines* (SVM). Con la investigación se demostró que es posible detectar la anemia, sin necesidad de un análisis de sangre, con mayor rapidez y con resultados confiables.

PALABRAS CLAVE: visión computacional / *machine learning* / anemia / detección / uñas

Comparison of Techniques Based on Computer Vision and Machine Learning for the Early Detection of Anemia From Nail Analysis

ABSTRACT. In Peru, anemia is a disease that affects more than 40% of the population, being common in both children and teenagers, and prevailing in pregnant women and children under 2 years of age, which seriously compromises their development. To diagnose anemia, it is necessary to perform blood tests to determine hemoglobin levels. However, most hospitals do not have the proper equipment to conduct the tests, which causes delays in the delivery of diagnoses. The objective of this research is to compare techniques based on computer vision and machine learning for the early detection of anemia from nail analysis, so that doctors can use such analysis as support in the early detection of this disease. With a timely diagnosis, it will be possible to prevent patients from suffering the disease in its different stages, especially for those who are at a chronic stage where the consequences are serious, since anemia can be a sign of an underlying disease. Several experiments were carried out and the best results were the following: accuracy 0.989, precision 0.98, recall 0.98 and F1-score 0.98. VGG19 architecture was used as a feature extractor in combination with the support vector machine (SVM) classifier. The research has shown that anemia can be detected without a blood test with quicker and reliable results.

KEYWORDS: computer vision / machine learning / anemia / detection / nails

1. INTRODUCCIÓN

La anemia es un problema de salud y nutrición pública. Se estima que más de dos mil millones de personas, es decir, un 30 % de la población mundial, tiene anemia en diferentes grados, datos que son revelados por la Organización Mundial de la Salud (OMS, 2008). En la OMS también se menciona que dicha enfermedad es muy común en niños y adolescentes, debido a que se presenta cuando disminuye el nivel de hemoglobina en la sangre por falta de hierro.

Según el Instituto Nacional de Estadística e Informática (INEI, 2018), el año 2017 se identificó que el 43,3 % de niños son anémicos y en el año 2018 se incrementó a 46,6 %, el cual equivale a 170 000 niños que padecen de esta enfermedad. En el informe se identificó que las gestantes también son una población vulnerable a dicha afección, comprometiendo el desarrollo embrionario y poniendo en riesgo la salud de la madre.

Para el descarte de anemia es necesario llevar a cabo pruebas de laboratorio como hemograma o análisis de sangre; estas son realizadas por especialistas que determinan los niveles de hemoglobina junto con otros parámetros que indican la morfología y tamaño de los hematíes (MedlinePlus, 2020). Para el descarte se tienen en cuenta los parámetros establecidos por la OMS, donde se detallan los valores normales de concentración de hemoglobina en niños, adolescentes, mujeres gestantes y puérperas. Por ejemplo, varones de 12 a 14 años tienen anemia cuando sus niveles de hemoglobina están por debajo de 11,9 gramos por decilitro (g/dl), mientras que los de 15 años a más tienen anemia cuando están por debajo de 13 g/dl (OMS, 2008).

En el Perú, las provincias alejadas de la capital carecen de hospitales que realicen el descarte de anemia. Por ejemplo, Madre de Dios, Ucayali y Tumbes solo cuentan con tres hospitales (INEI, 2018). Según el Ministerio de Salud (Minsa, 2015), en su mayoría, las provincias carecen de recursos humanos como médicos y enfermeras, y no cuentan con especialistas como cirujanos y hematólogos. Por lo tanto, el objetivo de esta investigación es comparar diferentes técnicas basadas en visión computacional y *machine learning* para la detección temprana de anemia a partir del análisis de uñas, con la finalidad de que se cuente con otras alternativas de descarte.

La anemia puede correlacionarse con la palidez en varias regiones anatómicas del cuerpo como los lechos ungueales, la conjuntiva y los pliegues palmares; cuanto más pálida sea la región, es más probable la presencia de la enfermedad (Thaver y Baig, 1998). Se decidió trabajar con imágenes de uñas de la mano de la investigación realizada por Sharma y Shrivastava (2015), donde se investigó que mediante el color y la textura de las uñas se pueden detectar enfermedades como la anemia y la diabetes. También está basada en Mannino, Myers, Tyburski, Caruso, Boudreaux, Leong, Clifford y Lam (2018), quienes afirman que es posible detectar la anemia mediante el análisis del color y los metadatos del lecho de las uñas y que tomar fotografías a la uña es más fácil, a diferencia de la conjuntiva, y tienen una baja variabilidad de tamaño y forma, a diferencia de los pliegues palmares.

Esta investigación se aplicó al contexto peruano y puede ser aplicado en otros contextos y en diferentes campos. La investigación se desarrolló en tres secciones; en primer lugar, se analizó la situación problemática; en segundo lugar, se desarrolló la metodología propuesta, los pasos a realizar son explicados de manera detallada con la finalidad de que el estudio pueda ser replicado. En tercer lugar, se analizaron los resultados y, finalmente, se presentaron las conclusiones y recomendaciones.

2. METODOLOGÍA

La metodología propuesta en la investigación consta de dos fases principales: construcción del *dataset* y preprocesamiento, de las que derivan tres escenarios (I, II y III) distintos donde se aplicarán las diferentes técnicas basadas en visión computacional y *machine learning*.

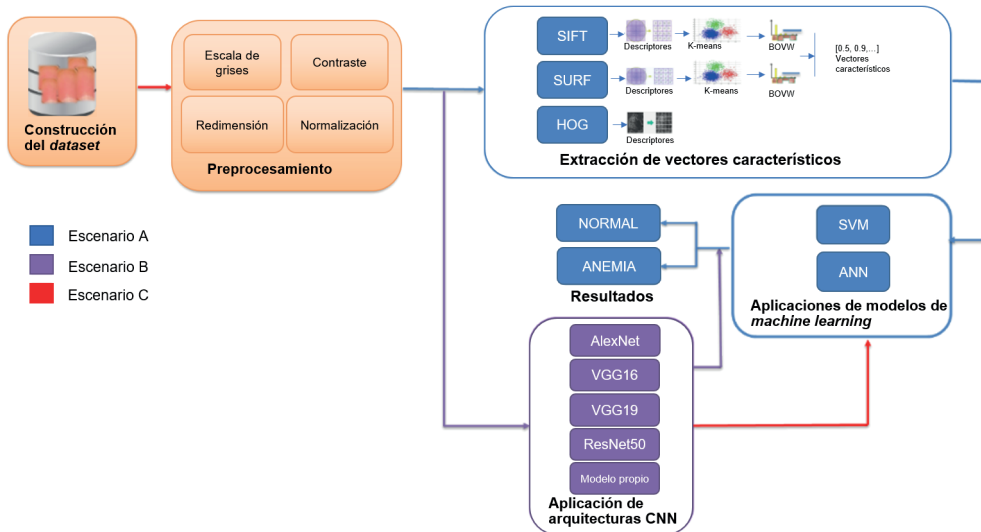


Figura 1. Metodología de la investigación
Elaboración propia

2.1 Construcción del *dataset*

Esta se realizó tomando fotos de uñas de personas con y sin anemia, estas fueron tomadas a una distancia de 5 cm. La distancia y las cantidades para el estudio se basaron en la investigación de Mannino *et al.* (2018). Todos los participantes se realizaron pruebas de laboratorio y con los resultados se etiquetaron las imágenes, esto con la finalidad de no tener errores al etiquetar entre personas sanas y con anemia (0,1), ya que a simple vista no es posible afirmar si la persona padece o no de la enfermedad.

El *dataset* contiene 2048 fotos de uñas, 1004 uñas sin anemia (que representan a 251 personas) y 1044 uñas con anemia (que representan a 261 personas). Por cada persona se recortaron cuatro fotos de las uñas, una de cada dedo (mano derecha), excluyendo el dedo pulgar porque, según estudios, no aporta información valiosa y las imágenes se deben recortar teniendo especial cuidado con el lecho ungueal que es la parte más importante de la uña. El recorte se realizó como se muestra en la figura 2. La conformación a detalle del *dataset* se muestra en la tabla 1.

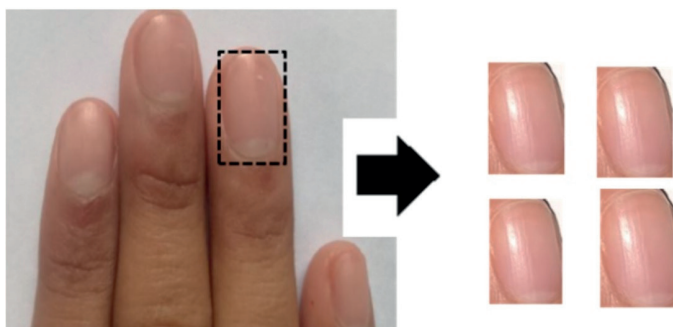


Figura 2. Ejemplo de recorte de las cuatro uñas
Elaboración propia

Tabla 1
Fotos de uñas con y sin anemia por género

Género	Sin anemia	Con anemia
Hombres (5-11 años)	30	31
Mujeres (5-11 años)	31	32
Hombres (12-17 años)	29	31
Mujeres (12-17 años)	28	29
Hombres (18-39 años)	32	33
Mujeres (18-39 años)	29	30
Hombres (40-80 años)	35	38
Mujeres (40-80 años)	37	37
Total	251	261

Elaboración propia

2.2 Preprocesamiento de datos

Esto se realizó en cuatro pasos: escala de grises, redimensión, contraste y normalización. Dichos pasos fueron necesarios, ya que las imágenes tienen diferentes tamaños y ruido.

2.2.1 Escenario A

En este escenario A (color azul de la figura 1) primero se realizan las dos fases (construcción del *dataset* y preprocesamiento), enseguida se realizarán las fases de extracción de vectores característicos y aplicación de modelos de *machine learning*.

2.2.1.1 Extracción de vectores característicos

En esta fase, las imágenes serán procesadas por tres métodos: Scale Invariant Feature Transformation (SIFT), Speeded-Up Robust Features (SURF) e Histogram of Oriented Gradient (HOG). El objetivo es obtener los vectores característicos, los cuales servirán como entradas para los modelos de clasificación.

- *Descriptor SIFT*: entre las principales salidas del descriptor, están los *keypoints*, que son las posiciones de los puntos característicos de la imagen (Lowe, 2004). Dichos descriptores tienen un tamaño de 128 y son enviados a los modelos de clasificación.
- *Descriptor SURF*: detectan los puntos de interés que son invariantes a la escala y rotación, pero de una manera mucho más rápida (Bay, Tuytelaars y Van Gool, 2006). El primer paso para su aplicación consiste en fijar una orientación reproducible basada en información de una región circular alrededor del punto de interés. Luego, se construye una región cuadrada alineada con la orientación seleccionada, y se extrae el descriptor SURF de ella, devolviendo una matriz de vectores, donde cada descriptor tiene 64 valores.

Como se describió, SIFT y SURF detectan sus puntos característicos de manera distinta, como se muestra en la figura 3. Y es importante mencionar que tienen como salida una matriz de vectores que tienen diferentes tamaños y se tiene que realizar dos operaciones adicionales para disminuir el tamaño de cada matriz: (i) técnica con base en la bolsa de palabras visuales (BOVW) y (ii) utilización de algoritmos de agrupamiento K-means donde cada centroide se utiliza como vocabulario del diccionario visual. Se realizaron experimentos con ($k=10, 50, 100, 200$).

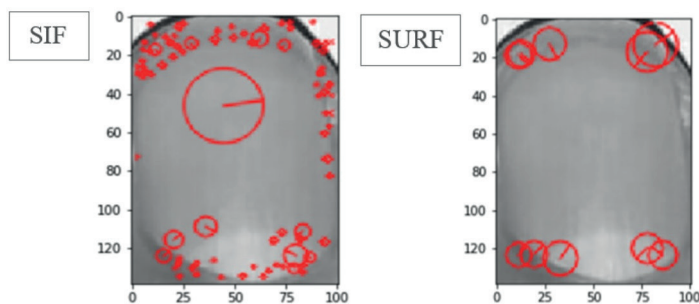


Figura 3. Ejemplo de los descriptores SIFT y SURF aplicados a una uña
Elaboración propia

- *Descriptor HOG*: fue utilizado para la detección de información de bordes y texturas de la imagen. Según Banerji, Sinha y Liu (2013), para la implementación del descriptor se siguieron diferentes pasos, como fraccionar las imágenes en distintas celdas para luego agrupar la orientación de gradientes y bloques que sirven como base para la normalización de los histogramas que representan el descriptor de la imagen que son los *inputs* para los modelos. El tamaño del vector depende de la cantidad de características; para la investigación se utilizó un tamaño de 64×128 , el cual da como resultado 3780.

2.2.1.2 Aplicación de modelos de *machine learning* (ML)

Se implementaron dos técnicas de *machine learning*: SVM y redes neuronales artificiales (ANN). Para los experimentos, el *dataset* se dividió en tres subconjuntos: 70 % para entrenamiento, 20 % para validación y 10 % para la prueba.

Para el caso de SVM, se utilizaron tres parámetros para realizar la búsqueda exhaustiva. Estos son: *kernel* (*linear*, *rbf* y *sigmoid*); el C que es el costo de la función y se tomaron valores de (1, 10, 100 y 1000); y, finalmente, el parámetro gamma con valores (0,1, 0,01, 0,001, 0,0001); la combinación se analizó con una validación cruzada (CV) de 5. Los parámetros fueron establecidos con base en estudios similares. En el caso de las arquitecturas ANN, se hizo la clasificación con tres arquitecturas y los mejores parámetros se muestran en la tabla 2.

Tabla 2
Mejores parámetros de cada arquitectura

Parámetros	ANN1	ANN2	ANN3
Capas ocultas	3	3	3
Número de neuronas en la primera capa oculta	100	10	100
Número de neuronas en la segunda capa oculta	50	50	25
Número de neuronas en la tercera capa oculta	25	25	10
Función de activación	sigmoide	sigmoide	sigmoide

Elaboración propia

2.2.1.3 Resultados del escenario A

En la tabla 3, se presentan los resultados con los tres descriptores (SIFT, SURF y HOG) y los algoritmos (SVM, ANN); se utilizaron k (k=10, 50, 100, 200) agrupaciones con las que se realizaron los experimentos.

En esta tabla se puede visualizar que el extractor de características SIFT, en combinación con el algoritmo SVM, tienen los mejores resultados tomando en cuenta las diferentes métricas, debido a que se trata de un tema de salud y se busca la mejor confiabilidad.

Tabla 3
Resultados de SIFT, SURF (k=10, 50, 100 y 200) y HOG

K	Descriptores	SIFT				SURF				HOG
		10	50	100	200	10	50	100	200	-
SVM	Accuracy	0,72	0,78	0,82	0,83	0,74	0,78	0,82	0,82	0,76
	Precision	0,75	0,77	0,82	0,83	0,77	0,97	0,82	0,85	0,81
	Recall	0,70	0,70	0,83	0,84	0,71	0,70	0,83	0,80	0,70
	F1-SCORE	0,72	0,78	0,82	0,83	0,74	0,78	0,82	0,82	0,79
ANN1	Accuracy	0,74	0,0	0,80	0,80	0,70	0,70	0,80	0,80	0,63
	Precision	0,79	0,82	0,85	0,85	0,73	0,82	0,85	0,85	0,62
	Recall	0,70	0,75	0,72	0,72	0,62	0,75	0,72	0,72	0,63
	F1-Score	0,73	0,78	0,78	0,78	0,67	0,78	0,78	0,78	0,62
ANN2	Accuracy	0,74	0,79	0,76	0,76	0,70	0,79	0,76	0,76	0,61
	Precision	0,79	0,73	0,77	0,77	0,70	0,73	0,77	0,77	0,62
	Recall	0,73	0,74	0,76	0,76	0,69	0,74	0,76	0,76	0,60
	F1-Score	0,72	0,72	0,78	0,78	0,71	0,72	0,78	0,78	0,61

(continúa)

(continuación)

ANN3	Accuracy	0,71	0,74	0,80	0,80	0,70	0,72	0,75	0,75	0,61
	Precision	0,72	0,75	0,80	0,80	0,70	0,73	0,74	0,74	0,62
	Recall	0,70	0,74	0,81	0,81	0,69	0,71	0,71	0,71	0,60
	F1-Score	0,72	0,76	0,78	0,78	0,71	0,74	0,75	0,75	0,61

Elaboración propia

2.2.2 Escenario B

En este escenario (color morado de la figura 1), primero se realizan las dos fases (construcción del *dataset* y preprocesamiento), enseguida se realiza la fase de aplicación de arquitecturas de las redes neuronales convolucionales (CNN). En este caso se utilizan las CNN para hacer la clasificación de las imágenes.

2.2.2.1 Aplicación de arquitecturas CNN

Para la investigación se utilizaron: ResNet50, VGG19, VGG16, AlexNet y también se propuso un modelo propio que ha sido inspirado en la arquitectura AlexNet. Para su aplicación se realizó *data augmentation* al *dataset* de entrenamiento (70 %), esto con el fin de aumentar la cantidad de imágenes y de esa manera se pueda entrenar las arquitecturas CNN. Por cada imagen del *dataset* en el entrenamiento, se incrementaron tres imágenes adicionales (aumento de brillo, filtros con la mediana y media). Esto debido a que estas tres variantes de la original permiten que la parte importante (lecho ungual) de la uña permanezca entera y sin rotar. También se realizó *transfer learning* en el cual se cargaron los pesos a las CNN, excepto del modelo propio.

En las CNN, también se tienen que optimizar parámetros (tabla 4), ya que no existen parámetros óptimos establecidos porque depende del tipo de datos con las que se trabaje. En esta investigación se optimizaron los parámetros en base a prueba y error. Por ejemplo, con respecto al número de *epochs*, una buena pista es incrementar el número hasta que la métrica *accuracy* con los datos de validación empieza a decrecer; incluso cuando el *accuracy* de los datos de entrenamiento continúe incrementándose, es cuando detecta un potencial sobreajuste u *overfitting* (Torres, 2018).

En la tabla 4, se muestran los mejores parámetros encontrados luego de evaluar cada arquitectura CNN, estas arquitecturas están compuestas por bloques de convolución que se alternan entre convolución y *subsampling* (Haykin, 2008), como se muestra en la figura 4.

Tabla 4
Parámetros modificados al momento de compilar

Modelos	Learning rate	Epochs	Batch_size
AlexNet	0,0000001	120	32
VGG16	0,00001	120	64
VGG19	0,00001	120	64
ResNet	0,000001	120	32
Modelo propio	0,00001	120	32

Elaboración propia

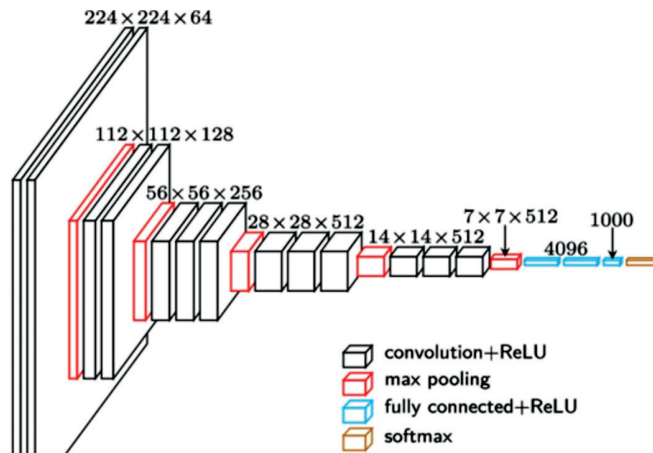


Figura 4. Arquitectura VGG16
Fuente: Frossard (2016)

2.2.2.1 Resultados del escenario B

De la tabla 5, se puede apreciar que los mejores modelos son de la familia Visual Geometry Group (VGG), siendo VGG19 quien tiene el resultado superior en términos de *accuracy*; al comparar la métrica *loss*, se puede ver que es menor en comparación a los demás modelos. Por tanto, el clasificador VGG19 tiene una mejor precisión y rendimiento del modelo.

Tabla 5
Resultados con las arquitecturas CNN

Arquitecturas	Accuracy	Precision	Recall	F1-score	Log Loss
VGG19	0,97	0,97	0,99	0,98	0,05
VGG16	0,96	0,96	0,97	0,96	0,06
AlexNet	0,94	0,95	0,90	0,92	0,18
RestNet50	0,96	0,96	0,98	0,97	0,07
Modelo propio	0,94	0,94	0,90	0,92	0,25

Elaboración propia

2.2.3 Escenario C

En este escenario (color rojo de la figura 1), primero se realizan las dos fases (construcción del *dataset* y preprocesamiento), luego se realizan las fases de aplicación de arquitecturas CNN y aplicación de modelos de *machine learning*. En este escenario, se usa la aplicación de arquitecturas CNN preentrenadas para obtener los vectores característicos y al final se usan los modelos de *machine learning* como clasificadores.

2.2.3.1 Aplicación de arquitecturas CNN

En esta investigación se propone la extracción de vectores característicos de la penúltima capa (capa densa) del modelo CNN preentrenados. Se extraen los vectores obtenidos por los filtros (*kernels*) de las CNN, los que sirven como *inputs* o entradas para los modelos de clasificación. Para este paso también se realizó *data augmentation* al *dataset*, con las mismas características y cantidades que el escenario anterior; para ello, se hizo uso de la RAM del Google Colab para que el proceso se realice en menor tiempo, esto debido a la cantidad de imágenes que se tienen que procesar.

El modelo previamente guardado con extensión h5 fue cargado y se extrajeron los vectores característicos de la penúltima capa del modelo CNN; luego, fueron enviados a los clasificadores de *machine learning* (como en el escenario A); el paso final de este tipo de escenario es el mismo que el de A, ya que se tienen que optimizar los parámetros de los modelos de *machine learning*.

2.2.3.2 Resultados del escenario C

De la tabla 6, se puede apreciar que los modelos que tienen resultados superiores son VGG19 y VGG16, los dos con los modelos SVM. Además, se puede observar que, con esta combinación, los modelos de *machine learning* tienen mejores resultados, en comparación con el escenario A.

Tabla 6
Resultados de las CNN de los modelos de machine learning

CNNS	Modelos ML	Accuracy	Precision	Recall	F1-score
VGG19	SVM	0,9890	0,98	0,98	0,98
	ANN	0,9589	0,95	0,94	0,94
VGG16	SVM	0,9821	0,98	0,98	0,98
	ANN	0,9581	0,95	0,95	0,95
AlexNet	SVM	0,9789	0,97	0,97	0,97
	ANN	0,9480	0,94	0,93	0,93
ResNet50	SVM	0,9850	0,98	0,98	0,98
	ANN	0,9570	0,95	0,96	0,96

Elaboración propia

3. RESULTADOS

Los resultados fueron obtenidos de los experimentos de tres (3) escenarios (A, B y C) y han sido evaluados con cuatro (4) métricas: *accuracy*, *precision*, *recall* y *F1-score*. De manera que, al realizar la comparación de estos, resulta que para este tipo de investigación se obtuvieron mejores resultados aplicando las técnicas CNN como extractores de características en combinación con las SVM, tal como se describe en el escenario C.

Se obtienen los mejores resultados con la VGG19 tanto en el escenario B como en el C, pero los resultados superiores se encuentran en el escenario C, donde se combinaron las CNN y los modelos SVM, y los resultados fueron superiores: *accuracy* 0,989, *precision* 0,98, *recall* 0,98 y *F1-score* de 0,98. Por lo que las mejores técnicas son las CNN junto con la arquitectura VGG19.

Con la aplicación de técnicas de visión computacional como SIFT, SURF y HOG se obtienen resultados inferiores a 95 %, por lo que dichas técnicas no son las adecuadas para este tipo de estudio porque las pruebas de laboratorio superan ese porcentaje de confiabilidad. Y lo que se pretende es tener resultados iguales o superiores que los métodos tradicionales, debido a que el resultado de la prueba involucra a la salud de las personas.

4. CONCLUSIONES

Se concluye que es posible detectar la anemia mediante el análisis de uñas empleando visión computacional y *machine learning*; también se identificó que las mejores técnicas para esta

investigación son las CNN y SVM. Como una contribución de la investigación, se demostró que es posible realizar el descarte de anemia sin la necesidad de un análisis invasivo, de manera rápida y fácil.

Se recomienda utilizar una mayor cantidad de imágenes para el *dataset* y de forma más eficiente, adoptar la propuesta e implementarla para la detección de otras enfermedades y en otros contextos; así se logrará menguar el número de personas con enfermedades en etapas avanzadas por la falta de diagnósticos tempranos.

REFERENCIAS

- Banerji, S., Sinha, A., y Liu, C. (2013). *HaarHOG: Improving the HOG Descriptor for Image Classification* [Sesión de conferencia]. IEEE International Conference on Systems, Man, and Cybernetics (pp. 4276-4281). 10.1109/SMC.2013.729
- Bay H., Tuytelaars T., Van Gool L. (2006). SURF: Speeded Up Robust Features. *Computer Science*, 3951. https://doi.org/10.1007/11744023_32
- De Benoist, B., Cogswell, M., Egli, I., y McLean, E. (2008). *Worldwide Prevalence of Anaemia 1993-2005*. WHO Global Database of Anaemia.
- Frossard, D. (2016). VGG en TensorFlow. [Figura]. <https://www.cs.toronto.edu/~frossard/post/vgg16/>
- Haykin, S. (2008). *Neuronal Networks and Learning Machines*. Pearson Prentice Hall. https://cours.etsmtl.ca/sys843/REFS/Books/ebook_Haykin09.pdf
- Instituto Nacional de Estadística e Informática. (2018). *Resultados de la Encuesta Demográfica y de Salud Familiar-ENDES 2018*. <https://www.ipe.org.pe/portal/wp-content/uploads/2018/06/INEI-Encuesta-Demografica-y-de-Salud-Familiar-ENDES.pdf>
- Instituto Nacional de Estadística e Informática. (2018b). Pobreza y gasto social. <https://www.inei.gob.pe/estadisticas/indice-tematico/sociales/>
- Lowe, D. (2004). Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision*, 60(2), 91-110. 10.1023/B:VISI.0000029664.99615.94
- Mannino, R. G., Myers, D. R., Tyburski, E. A., Caruso, C., Boudreaux, J., Leong, T., Clifford, G. D., y Lam, W. A. (2018). Smartphone app for non-invasive detection of anemia using only patient-sourced photos. *Nature Communications*, 9(1), 4924. <https://doi.org/10.1038/s41467-018-07262-2>
- MedlinePlus. (6 de agosto del 2020). *Anemia*. *National Library of Medicine*. <https://medlineplus.gov/anemia.html>

- Ministerio de Salud. (2015). *Necesidad de médicos especialistas en establecimientos de salud del Sector Salud, Perú 2010*. <http://bvs.minsa.gob.pe/local/MINSA/1613.pdf>
- Sharma, V., y Shrivastava, A. (2015). System for Disease Detection by Analyzing Finger Nails Color and Texture. *International Journal of Advanced Engineering Research and Science*, 5.
- Thaver, I. H., y Baig, L. (1994). Anaemia in Children: Part I. Can Simple Observations by Primary Care Provider Help in Diagnosis? *Journal of Pakistan Medical Association* 44, 282-284.
- Torres, J. (2018). *Deep Learning. Introducción práctica con Keras* (1.ª ed.). <https://torres.ai/deep-learning-inteligencia-artificial-keras/>