

CONTEO DE GRÁNULOS USANDO REDES NEURONALES TIPO U-NET Y COMPONENTES CONECTADOS

JUAN FELIPE MONSALVO SALAZAR
<https://orcid.org/0000-0002-9450-2401>
Organización Corona, Medellín, Colombia

JUAN RODRIGO JARAMILLO POSADA
<https://orcid.org/0000-0001-8057-5274>
Adelphi University, Facultad de Negocios, Nueva York, Estados Unidos

Recibido: 2 de junio del 2021 / Aprobado: 11 de junio del 2021

doi: <https://doi.org/10.26439/ing.ind2022.n.5804>

RESUMEN. Este trabajo desarrolla una metodología para automatizar el conteo de gránulos remanentes en una taza sanitaria (prueba ASME A112.19.2-2018/CSA B45.1-18). Esta metodología integra una red convolucional U-Net que fue entrenada con 3678 imágenes y una variación del algoritmo de componentes conectados. Los resultados arrojan una precisión superior al 98 % para valores entre 0 y 180 gránulos. La metodología se implementó en la línea de producción.

PALABRAS CLAVE: inteligencia artificial / visión por computadora / redes neuronales / automatización

COUNTING GRANULES WITH U-NET NETWORKS AND CONNECTED COMPONENTS

ABSTRACT. This research develops a methodology to automate the process of counting the number of granules that remains in a toilet after being flushed (ASME A112.19.2-2018/CSA B45.1-18). This work integrates a U-Net convolutional network with a variation of the connected component algorithm. The training set consisted of 3678 images. Results show an accuracy above 98% between 0 and 180 granules. The methodology has been implemented in the production line.

KEYWORDS: artificial intelligence / computer vision / neural networks / automation

Correos electrónicos en orden de aparición: jmonsalvo@corona.com.co, jjaramillo@adelphi.edu

INTRODUCCIÓN

Existen muchas aplicaciones en las que se requiere el conteo de elementos, como los cultivos de bacterias, el manejo de multitudes o el conteo de unidades producidas (manufactura) o recolectadas (agricultura). El conteo de unidades puede ser muy dispendioso debido a la cantidad de elementos involucrados, la frecuencia requerida y los errores. En muchos casos, la automatización de dichos procesos es compleja por la dificultad para organizar los elementos por contar, como ocurre con los cultivos de bacterias o con las personas presentes en una multitud. En estos casos, el conteo utilizando visión artificial se convierte en una alternativa con gran potencial.

Las primeras aplicaciones para el conteo de objetos con visión artificial se basan en la identificación de objetos similares con base en sus colores. Uno de los primeros métodos para contar objetos en imágenes es la transformación de los círculos de Hough (Hough, 1962), que identifica los bordes de objetos circulares en imágenes. Este método solo funciona bien para objetos redondeados. Otro de los métodos básicos es el de los componentes conectados, que reconoce los elementos presentes en una imagen mediante la identificación de los píxeles del mismo color que se conectan de forma horizontal, vertical o diagonal. Los resultados de estos métodos son muy sensibles a los cambios de color generados por los cambios de luz, la inclinación y distancia de la cámara, el contacto entre objetos similares, etcétera. Dichos resultados se pueden mejorar ligeramente convirtiendo las imágenes a tonos grises o solo blanco y negro. La librería OpenCV (Bradski, 2000) contiene los algoritmos para generar círculos de Hough y componentes conectados. Desafortunadamente, los márgenes de error generados con estos métodos son muy altos para implementaciones en el sector real.

De otro lado, el avance en la velocidad de cálculo en los procesadores y en las unidades de procesamiento gráfico ha potenciado el desarrollo de las redes neuronales. Entre los adelantos de las redes neuronales, destacan las redes neuronales convolucionales (Fukushima, 1980), las cuales se utilizan para el análisis de imágenes y fotografías, entre muchas otras aplicaciones (Shamsaldin et al., 2019). El uso de las redes convolucionales permite la aplicación de filtros para procesar las imágenes y lograr la identificación de objetos por medio de otras técnicas como las mencionadas anteriormente (círculos de Hough y componentes conectados) o los mapas de densidad (también denominados mapas de calor). La aplicación de múltiples filtros hace posible la segmentación de las imágenes, que conduce a simplificar las imágenes removiendo fondos y objetos que no son de interés para facilitar su interpretación. Una de las variantes de las redes convolucionales son las redes U-Net (Ronneberger et al., 2015), las cuales presentan muy buen desempeño en la segmentación de imágenes, con múltiples aplicaciones en el campo de la radiología, así como en la generación de mapas de densidad.

Las redes convolucionales tienen un rol preponderante en las aplicaciones de la visión artificial al conteo de elementos presentes en imágenes. Para el conteo de grandes cantidades de objetos en una misma imagen, como en cultivos, automóviles y multitudes, la aproximación que ofrece mejores resultados hasta el momento es la utilización de redes convolucionales para generar mapas de densidad. Estos mapas se procesan utilizando modelos de regresión para estimar el número de objetos en la imagen original. Algunas de las aplicaciones más recientes en esta dirección son los trabajos de Wu et al. (2019) para el conteo de plántulas de arroz y de Zhao et al. (2018) para el conteo de multitudes. Wu et al. (2019) utilizan drones que toman fotografías a color, las cuales son procesadas utilizando una red convolucional para generar mapas de densidad a los que se les aplica una regresión para estimar el número de plántulas. El modelo se entrenó usando 40 fotografías con números de plántulas entre 3732 y 16 173. La precisión obtenida por el modelo fue de aproximadamente el 93 %. Por su parte, Zhao et al. (2018) emplean una red convolucional restringida para generar mapas de densidad, acompañada de una función de pérdida reticular. Esta técnica permite el mejoramiento de los mapas en cada convolución. Luego se le aplica un modelo de regresión al mapa de densidades final para estimar el número de objetos en la multitud. Los resultados presentados por los autores superan otros métodos existentes en la literatura utilizando el set de imágenes anotadas ShanghaiTech partes A y B. Finalmente, Ilyas et al. (2020) presentan una revisión detallada del conteo de multitudes, en la cual se resalta el éxito de las diferentes variaciones de redes convolucionales para generar mapas de densidad más precisos, y plantean la integración de máquinas de Boltzmann y redes convolucionales como una de las estrategias más prometedoras hacia el futuro.

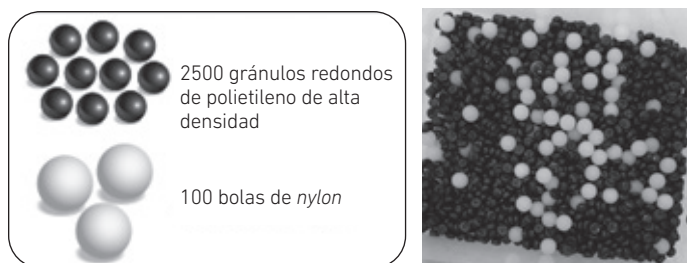
Otra área importante es el conteo de células y bacterias, que se realiza en diagnósticos médicos e investigaciones biológicas. Tradicionalmente, este conteo se hace de forma manual con la ayuda de equipos de laboratorio o componentes químicos. Por su parte, Alam e Islam (2019) utilizan la red preentrenada YOLO (*You Only Look Once*) para identificar y contar tres clases diferentes de células sanguíneas (glóbulos rojos, glóbulos blancos y plaquetas) con una precisión del 96 %. Otra aplicación interesante es descrita en Yong et al. (2019) para detectar y contar insectos nocivos en los cultivos para la producción de aceite en China. El estudio evaluó tres conocidos modelos de la literatura (R-CNN, R-FCN y SSD). Los autores reportan resultados superiores al modelo tradicional ResNet-50 con una precisión del 77,14 %. Finalmente, Dobrescu et al. (2019) plantean una aplicación para contar el número de hojas en plantas. Los autores utilizaron el modelo VGG16 (red neuronal convolucional) y la entrenaron con la técnica de propagación relevante. Dicha red produce un mapa de densidades que alimenta una red neuronal tradicional de tres capas con 1024, 512 y un nodo respectivamente. Los resultados obtenidos superan los del modelo de redes convolucionales ResNet-50.

Descripción del problema

Corona es una empresa multinacional con sede en Colombia que se especializa en la fabricación de productos cerámicos; su división de baños y cocina se enfoca en el desarrollo y fabricación de porcelana sanitaria (lavamanos, sanitarios, pedestales). Al ser una empresa multinacional con exportaciones a toda América, debe cumplir con diferentes estándares de calidad y desempeño que certifican la calidad del producto comercializado. Uno de ellos es la norma americana ASME A112.19.2-2018/CSA B45.1-18, "Ceramic Plumbing Fixtures", la cual especifica las pruebas y desempeño que deben cumplir los sanitarios. En el numeral 7.5 se encuentra la prueba de gránulos (*granule and ball test*) que evalúa el manejo de cargas livianas; esta prueba consiste en agregar 2500 gránulos redondos de polietileno de alta densidad (PEAD), de un diámetro aproximado de 4 mm, y 100 bolas de *nylon*, de un diámetro aproximado de 6,35 mm, en el pozo del sanitario. La figura 1 muestra imágenes de los gránulos y bolas usados en la prueba.

Figura 1

Gránulos para la prueba ASME A112.19.2-2018



La prueba requiere descargar el sanitario y, después, contar los gránulos y bolas visibles que quedan en el pozo. El ensayo se considera satisfactorio si, al ejecutar la prueba tres veces seguidas, en ninguna de las ocasiones quedan visibles más de 125 gránulos o 5 bolas de *nylon* en el pozo del sanitario. La prueba, a pesar de ser sencilla de realizar, es un cuello de botella dentro del plan de calidad de evaluación de los productos, ya que el hecho de realizar el conteo de los gránulos de manera manual por operarios del laboratorio puede llegar a consumir un periodo de tiempo considerable (10 minutos) y más si se considera que la prueba debe efectuarse tres veces por cada sanitario evaluado. Por eso, surge la oportunidad de mejorar este proceso de evaluación al implementar una herramienta de visión artificial que realice el conteo de forma automática, precisa y en un menor tiempo.

MATERIALES Y MÉTODOS

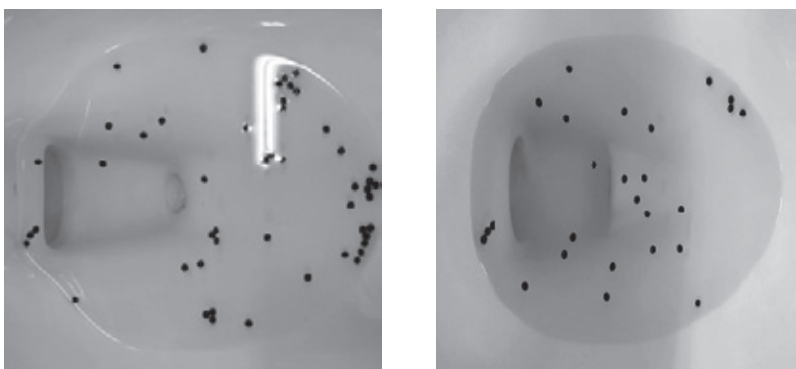
Para estimar el número de gránulos remanentes en la taza durante la prueba ASME A112.19.2-2018/CSA B45.1-18, se desarrolló un algoritmo de componentes conectados para analizar imágenes segmentadas con una red convolucional del tipo U-Net. Para desarrollar el algoritmo, se siguieron estos pasos:

1. Recolección de información
2. Segmentación de imágenes
3. Entrenamiento de la red U-Net
4. Ajuste del algoritmo de componentes conectados
5. Evaluación de la metodología en el campo
6. Implementación en la línea de producción

Inicialmente, se recolectaron videos simulando los resultados de la prueba con remanentes de gránulos en el espejo de agua entre 1 y 180 unidades; estos videos se tomaron para diferentes referencias de tazas sanitarias bajo múltiples condiciones de iluminación. La duración de cada video es de aproximadamente 30 segundos. Es importante anotar que cada imagen es ligeramente distinta, ya que en los videos aparecen los gránulos en movimiento. Los videos se procesaron para obtener varias imágenes de cada configuración grabada para un total de 3678 imágenes. La figura 2 muestra imágenes de dos videos diferentes.

Figura 2

Imágenes de cada configuración grabada



El siguiente paso del proceso consistió en segmentar las imágenes. El proceso de segmentación consiste en remover el fondo de la imagen dejando solamente los gránulos. Dado el número de imágenes disponibles (3678 imágenes), hacer la segmentación de

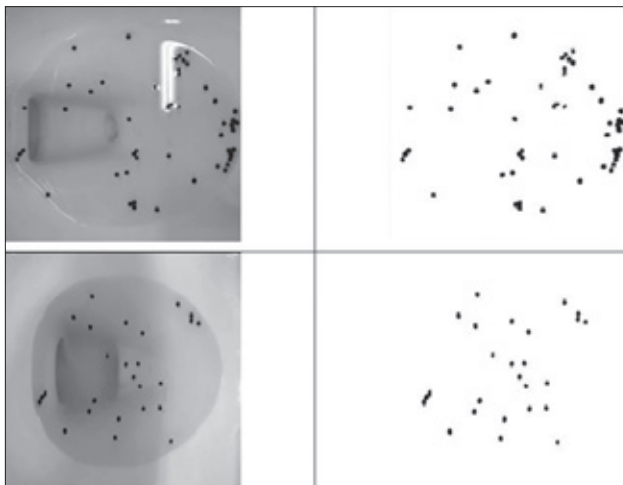
forma manual no es práctico. Afortunadamente, las imágenes digitales se representan como matrices. Una imagen a color se constituye con tres matrices, para los niveles de rojo, verde y azul (RGB). Como en este caso los colores no son importantes, las imágenes se pueden convertir a tonos de gris, lo cual elimina dos matrices. Una forma de segmentar las imágenes en gris es aplicar un filtro a la imagen. Los filtros se representan como matrices y la operación para aplicar el filtro se denomina una convolución. Luego de múltiples experimentaciones con diferentes filtros, se encontró que el filtro H (figura 3) genera resultados aceptables. El valor óptimo de K varía en función de la luz en la imagen.

Figura 3
Filtro H

$$\begin{array}{|c|c|c|} \hline -1 & 0 & -1 \\ \hline -1 & K & -1 \\ \hline -1 & 0 & -1 \\ \hline \end{array}$$

Al utilizar el filtro H, se hace necesario calibrar el valor de K para cada video. Después de algunos análisis, se encontró que el valor óptimo de K varía entre 7,5 y 12,5. La figura 4 muestra los resultados obtenidos al aplicar el filtro a las imágenes de la figura 2.

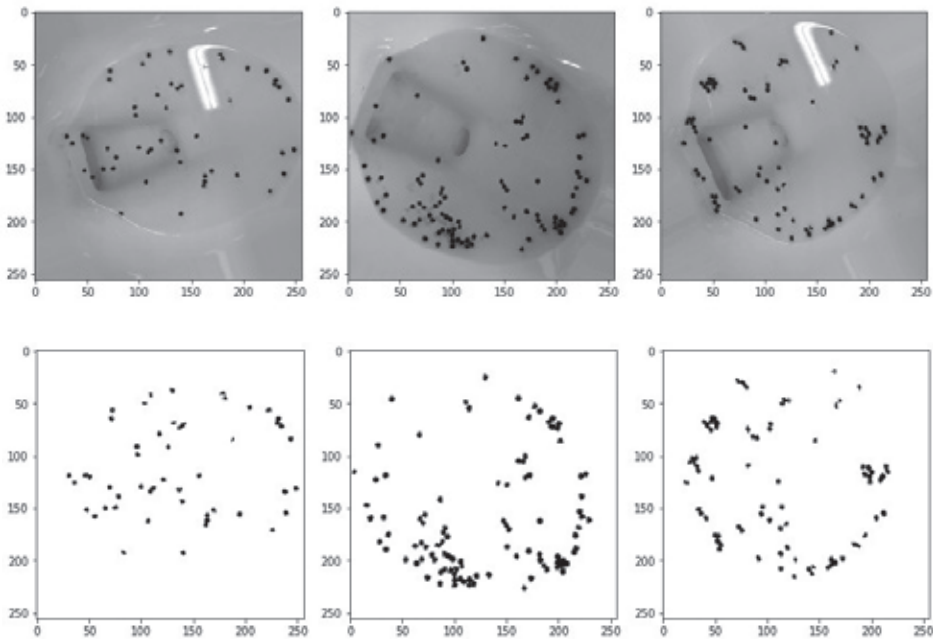
Figura 4
Imagen original (izquierda) y segmentación generada con el filtro H (derecha)



Desafortunadamente, calibrar el valor de K para cada imagen es ineficiente. Por este motivo, se incorporó una red U-Net para eliminar la sensibilidad de K a las diferentes condiciones de iluminación. Para entrenar la red U-Net se utilizaron 2869 imágenes con sus correspondientes segmentaciones (set para entrenamiento) y 809 imágenes para la validación (set para validación). Las segmentaciones fueron obtenidas aplicando el filtro H con valores de K controlados para garantizar la mejor segmentación. El conjunto imagen original e imagen segmentada del set de entrenamiento fue sometido a un proceso de distorsión (*data augmentation*) para abarcar mayores escenarios e incrementar el número de imágenes disponibles. Así, durante este proceso, las imágenes originales fueron transformadas aplicando rotaciones horizontales y verticales, desplazamientos en los dos ejes, así como acercamientos (véase la figura 5).

Figura 5

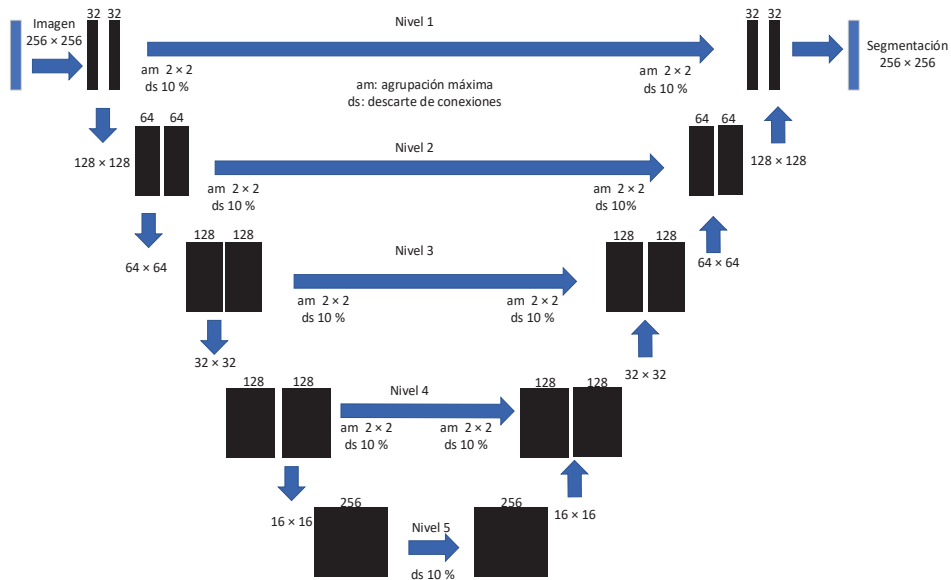
Distorsión de imágenes



La figura 6 muestra los detalles de la red U-Net utilizada en este estudio. La U-Net procesa imágenes redimensionadas a 256×256 píxeles, como las que aparecen en la parte izquierda de la figura 4, y genera imágenes como las que se muestran en la parte derecha de la figura 4.

Figura 6

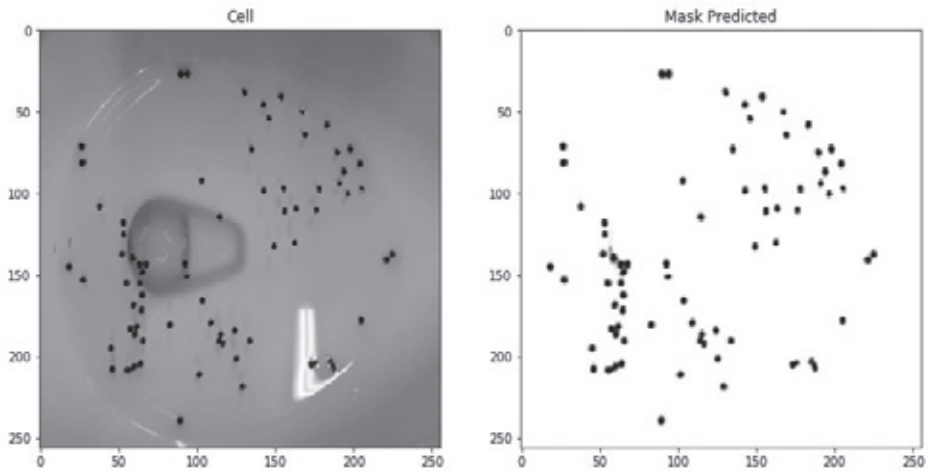
La red U-Net utilizada en este estudio



La U-Net cuenta con cinco niveles, comienza con una imagen y va generando más imágenes aplicando filtros convolucionales (similares al filtro H). Cada nivel contiene dos series de filtros convolucionales; por ejemplo, el nivel 1 contiene 32 filtros en cada serie. En cada nivel se aplica la agrupación máxima para reducir cada imagen en un 50 % y un descarte aleatorio de conexiones del 10 % para prevenir sobreajuste del modelo. La red U-Net, primero, genera una gran cantidad de imágenes utilizando los filtros convolucionales. Los tamaños de las imágenes se van reduciendo a la mitad hasta que la red alcanza el nivel 5. En ese momento, la red comienza a consolidar imágenes hasta llegar de nuevo al nivel 1, generando una nueva imagen. Una de las características más interesantes de la red U-Net es la retroalimentación que ocurre en cada nivel entre las imágenes generadas y las imágenes consolidadas. La U-Net, como todas las redes neuronales, requiere entrenamiento para procesar las imágenes. El proceso de entrenamiento genera los valores asociados a cada filtro.

Figura 7

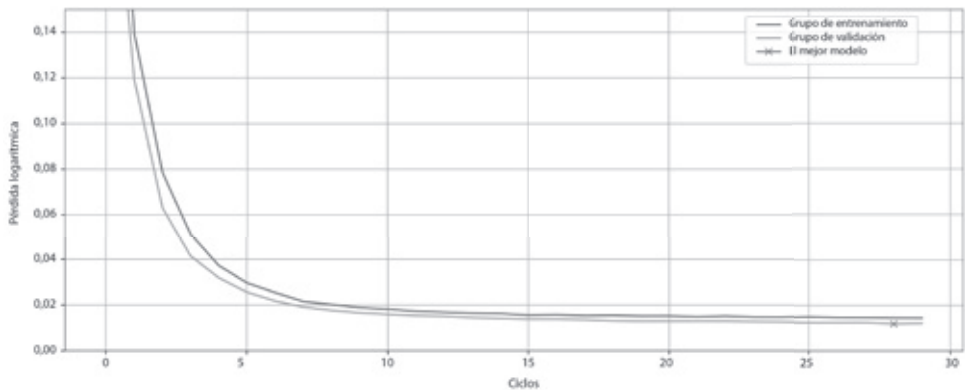
Resultados de la U-Net



La U-Net se entrenó durante 30 ciclos (*epochs*); cada ciclo consiste en alimentar todas las imágenes, generar las respectivas segmentaciones, comparar cada segmentación con la segmentación original y ajustar todos los valores asociados a los filtros en función de las diferencias (errores) entre las segmentaciones originales y las generadas por la U-Net. La figura 8 muestra la evolución del nivel de error durante el proceso de entrenamiento. El eje Y representa el error como pérdida logarítmica (*log loss*) y el eje X, los ciclos. La curva azul grafica el error para el grupo de entrenamiento y la curva naranja, el error para el grupo de validación. Es importante anotar que solo el error del grupo de entrenamiento se utiliza para el ajuste de la U-Net. La cruz roja representa el mejor modelo. El entrenamiento de la U-Net tardó aproximadamente 30 minutos.

Figura 8

Pérdida logarítmica (*log-loss*) de la U-Net



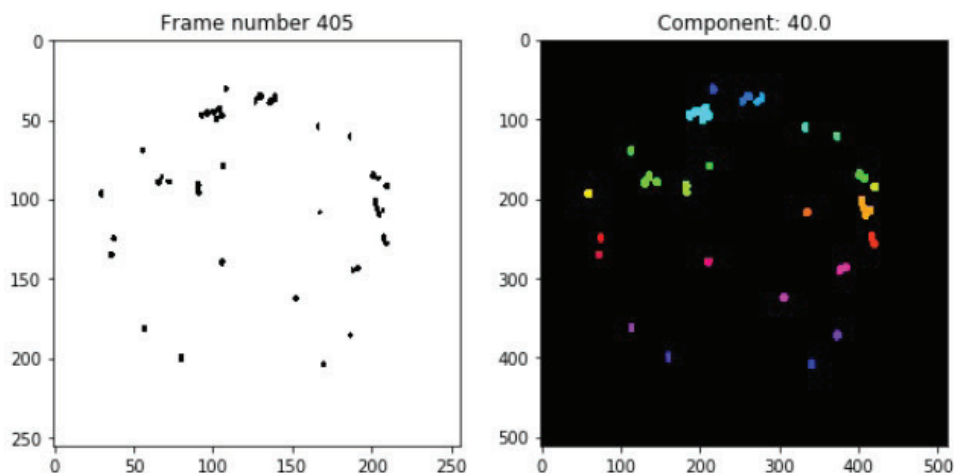
Al finalizar este proceso, se logra tratar la imagen recibida por el sistema de visión artificial eliminando toda información que no es importante para el conteo, de manera que queda una imagen donde se identifican claramente los gránulos y se elimina cualquier reflejo o forma del pozo. El componente final del modelo desarrollado es el algoritmo para estimar el número de gránulos que aparecen en la segmentación generada por la red U-Net. Para dicho conteo, se utiliza el algoritmo de componentes conectados, el cual cuenta el número de componentes presente en la imagen. El algoritmo detecta los píxeles negros que son contiguos para hallar los componentes. Algunos de los componentes contienen más de un gránulo. Cuando dos o más gránulos aparecen juntos, el algoritmo los reporta como un solo componente. Por este motivo, el algoritmo de componentes conectados se modifica de la siguiente manera:

1. Identificar cada componente
 - a. Definir cada componente con base en los píxeles contiguos del mismo color (verticales, horizontales o diagonales)
2. Obtener el número de píxeles para cada componente
3. Encontrar la moda para el número de píxeles de cada componente
4. Dividir el número de píxeles de cada componente por la moda
 - a. Redondear el valor obtenido
5. Sumar los valores obtenidos en el paso 4

La figura 9 muestra una imagen segmentada (izquierda) y los componentes detectados (derecha). En este caso, el número de gránulos obtenidos por el algoritmo de componentes conectados es de 40 unidades.

Figura 9

Componentes conectados



En resumen, la metodología utilizada para el conteo de los gránulos es de la siguiente manera:

1. Para cada imagen en el video:
 - a. Generar la segmentación con la U-Net
 - b. Aplicar el algoritmo modificado de componentes conectados
2. Promediar el número de gránulos encontrados para cada imagen

La red se programó en Python 3.7 utilizando Keras, Tensor Flow y OpenCV. La computadora que se utilizó tiene un procesador i7-8700K @ 3.7 GHz, una memoria RAM de 32 GB y una tarjeta gráfica NVIDIA GeForce GTX 1080 de 8 GB de memoria dedicada.

RESULTADOS

Para validar la precisión de la metodología, se recolectaron nuevos videos de forma controlada, garantizando diferentes niveles de iluminación y cantidades de gránulos representativas de la operación normal del proceso de control de calidad. Con el fin de recolectar los videos, se procedió de la siguiente manera:

1. Se seleccionan tres tazas distintas y tres ubicaciones distintas (variación de la luz).
2. Para cada taza y ubicación:
 - a. Se agregan cinco gránulos en el pozo de la taza.
 - b. Se utiliza el método desarrollado para contar los gránulos.
 - c. Se repiten los pasos a y b hasta llegar a un valor de 180 gránulos.
3. Se promedia el resultado de cada medición en los tres escenarios distintos.

La tabla 1 y la figura 10 muestran la precisión obtenida. En la figura 10, la línea azul es el dato real de gránulos en el pozo y la línea roja es el conteo realizado por el método desarrollado. Entre 0 gránulos y 100 gránulos, el error es del 0 %; entre 100 y 130 gránulos, el error es en promedio del 1 %; finalmente, entre 130 y 180 gránulos, se encuentra un error en promedio del 2 %, lo que representaría en el peor de los casos (180 gránulos) un desvío de ± 4 gránulos.

Tabla 1
Tabla de resultados

Dato real (gránulos)	Medición del sistema (gránulos)	Error (%)
5	5	0
10	10	0
15	15	0
20	20	0
25	25	0
30	30	0
35	35	0
40	40	0
45	45	0
50	50	0
55	55	0
60	60	0
65	65	0
70	70	0
75	75	0
80	80	0
85	84	1
90	90	0
95	95	0
100	99	1
105	105	0
110	109	1
115	114	1
120	118	1
125	124	1
130	128	2
135	132	2
140	137	2
145	141	3
150	146	3
155	152	2
160	158	1
165	162	2

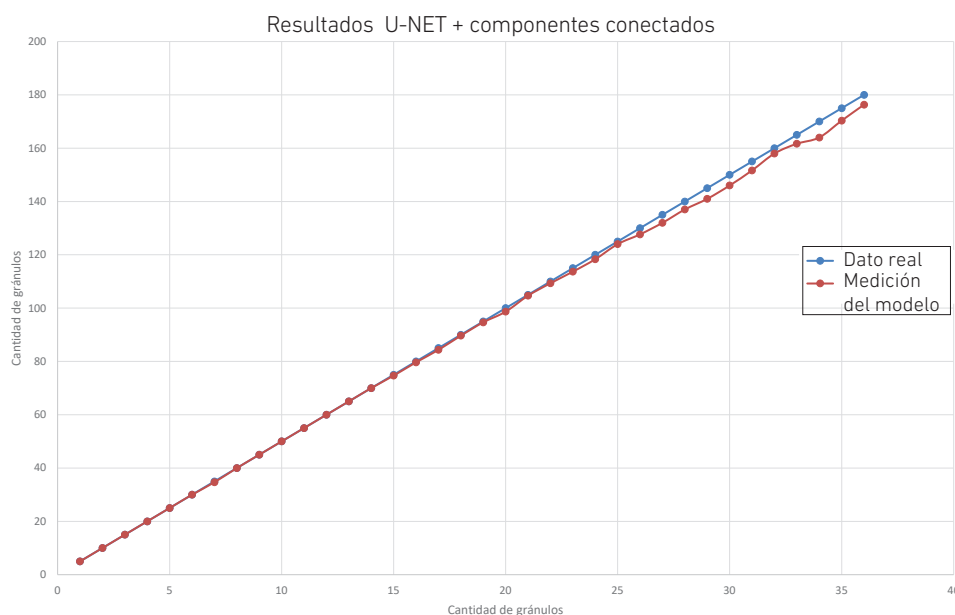
(continúa)

(continuación)

170	164	4
175	170	3
180	176	2

Figura 10

Precisión de la metodología



Estos resultados nos muestran que la precisión del método se ve afectada a medida que se aumenta la cantidad de gránulos en el pozo y la aglomeración de estos en clústeres, comportamiento asociado al proceso de dividir el número de píxeles de cada componente por la moda del número de píxeles. Este resultado es coherente con los obtenidos por otros investigadores, en los cuales los métodos para el conteo de aglomeraciones no funcionan bien con el conteo de elementos con bajas densidades y viceversa. La precisión obtenida está por encima de la alcanzada por otros métodos descritos en la literatura, como los mencionados anteriormente. Adicionalmente, la precisión alcanzada para el nivel crítico de la norma (125 gránulos) permite la implementación del método en el ambiente industrial garantizando la confiabilidad de los resultados.

Finalmente, el método implementado reduce el tiempo de ejecución de la prueba en un 90 %, ya que los resultados que arroja el método son instantáneos, agilizando de manera satisfactoria el cuello de botella de esta prueba (el conteo manual de los

gránulos). Asimismo, permite la creación de un archivo digital, eliminando el error humano y facilitando la trazabilidad.

Montaje y despliegue

Al lograr desarrollar una herramienta rápida y precisa en el conteo de gránulos, también se debe implementar un aplicativo cómodo e intuitivo que facilite el uso de esta herramienta, así como un montaje donde repose el sistema de visión artificial y que sea asequible rápidamente para cuando se requiera su uso. Con estas premisas se elaboró el montaje que se observa en la imagen de la derecha de la figura 11. Este montaje consiste en un sistema de manos libres donde reposa la cámara y que permite su posicionamiento tanto en el plano vertical como en el horizontal, brindando la posibilidad de abarcar las diferentes geometrías y tamaños presentes en las variadas referencias de sanitarios que se fabrican en Corona. Está acompañado de una interfaz gráfica que permite la interacción con el sistema de visión artificial (véase la figura 11).

Figura 11

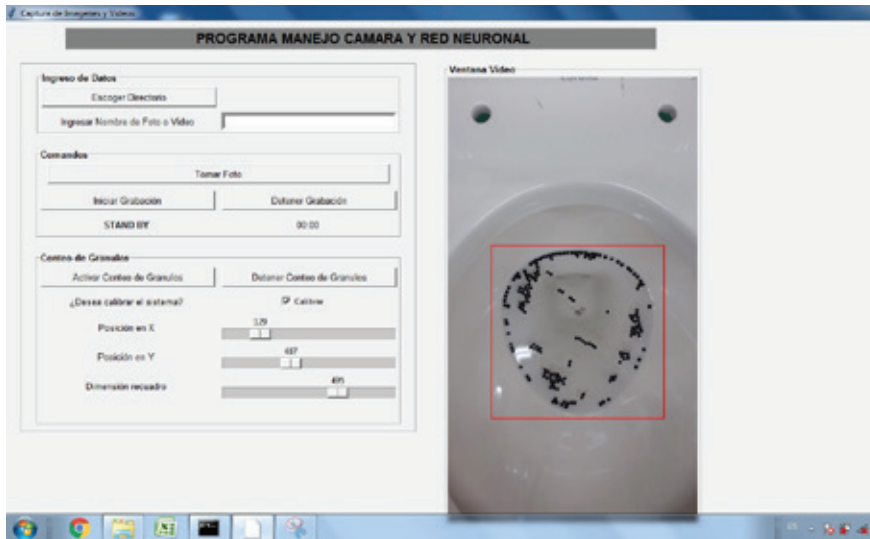
Montaje del sistema de visión artificial



La interfaz gráfica, que fue desarrollada en Python 3.7 usando la librería Tkinter, sirve para interactuar con la cámara y realizar grabaciones o fotografías del sanitario evaluado, con lo cual se puede generar una evidencia histórica de las pruebas realizadas. Igualmente, la interfaz permite correr la metodología desarrollada; para ello, se tiene un paso intermedio de calibración de la herramienta, ya que se necesita delimitar el espacio del video donde se realizará el conteo (véase la figura 12).

Figura 12

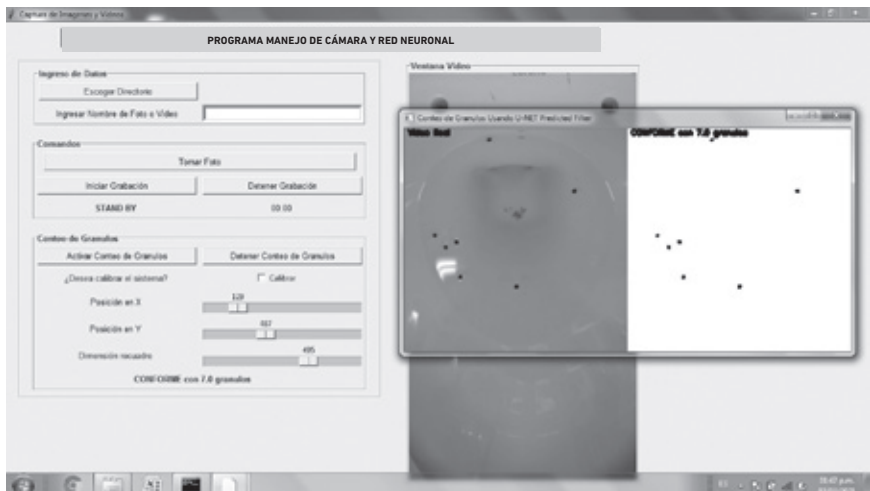
Interfaz de usuario y calibración de la metodología



Una vez que se realiza la calibración, se puede usar el botón de *activar conteo de gránulos* para hacer el llamado al método desarrollado. Esto activará una ventana emergente como la que se observa en la figura 13, donde se verá la imagen recibida por la U-Net y la segmentación creada por dicha red. En la parte inferior de la interfaz, aparecerá un mensaje donde se indica la totalidad de gránulos contados y si el resultado de dicho conteo se puede considerar como conforme o no, según la normatividad mencionada.

Figura 13

Aplicativo funcionando



CONCLUSIONES Y RECOMENDACIONES

La motivación de este trabajo es la agilización del proceso para aplicar la norma ASME A112.19.2-2018/CSA B45.1-18, "Ceramic Plumbing Fixtures", para la evaluación de sanitarios. En este sentido, se presenta una solución novedosa para el conteo de gránulos en imágenes utilizando redes neuronales en combinación con un algoritmo tradicional para automatizar la aplicación de la norma. En particular, este trabajo desarrolla un filtro (H), que permite el preprocesamiento de imágenes para el entrenamiento del algoritmo de forma eficiente; una red U-Net para la segmentación de las imágenes; e integra un algoritmo de componentes conectados con modificaciones para el conteo de gránulos. La precisión de los resultados obtenidos se compara positivamente con los reportados en la literatura para problemas similares. La metodología desarrollada es de fácil implementación en el ambiente industrial y disminuye el tiempo para la evaluación de una taza sanitaria en un 90 %. Actualmente, la solución reemplazó el conteo de gránulos de forma manual y está completamente integrada al proceso de producción.

REFERENCIAS

- Alam, M., & Islam, M. (2019). Machine learning approach of automatic identification and counting of blood cells. *Healthcare Technology Letters*, 6(4), 103-108. DOI: 10.1049/htl.2018.5098
- Bradski, G. (2000). The OpenCV Library. *Journal of Software Tools*, 25, 120-125.
- Dobrescu, A., Giuffrida, M., & Tsaftaris A. (2019). Understanding deep neural networks for regression in leaf counting. En *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)* (pp. 2600-2608). DOI: 10.1109/CVPRW.2019.00316
- Fukushima, K. (1980). Neocognitron: a self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological Cybernetics*, 36(4), 193-202. <https://doi.org/10.1007/BF00344251>
- Hough, P. (1962). *Method and means for recognizing complex patterns* (US Patent 3,069,654, Ser. N.º 17,7156 Claims).
- Ilyas, N., Shahzad, A., & Kim, K. (2020). Convolutional-neural network-based image crowd counting: review, categorization, analysis, and performance evaluation. *Sensors*, 20(1). <https://doi.org/10.3390/s20010043>
- Ronneberger, O., Fischer, P., & Brox, T. (2015). *U-Net: convolutional networks for biomedical image segmentation*. <https://arxiv.org/abs/1505.04597>

- Shamsaldin, A., Fattahb, P., Rashid, T., & Al-Salihi, N. (2019). A study of the convolutional neural networks applications. *UKH Journal of Science and Engineering*, 3(2), 31-40. <https://doi.org/10.25079/ukhjse.v3n2y2019.pp31-40>
- Wu, J., Yang, G., Yang, X., Xu, B., Han, L., & Zhu, Y. (2019). Automatic counting of in situ rice seedlings from UAV images based on a deep fully convolutional neural network. *Remote Sensing*, 11(6), 691. <https://doi.org/10.3390/rs11060691>
- Yong, H., Hong, Z., Yangyang, F., Shuaisheng, J., & Jianjian, W. (2019). Application of deep learning in integrated pest management: a real-time system for detection and diagnosis of oilseed rape pests. *Mobile Information Systems, 2019*, Artículo 4570808. <https://doi.org/10.1155/2019/4570808>
- Zhao, M., Zhang, J., Zhang, C., & Zhang, W. (2018). Towards locally consistent object counting with constrained multi-stage convolutional neural networks. En *Asian Conference on Computer Vision* (pp. 247-261). Springer. https://doi.org/10.1007/978-3-030-20876-9_16

