Detección y clasificación automática de levaduras de cerveza para el análisis de viabilidad

Noelia Falczuk[1], Luna Sanes[1], Pablo Negri[1 2], Clara Bruzone[3], and Diego Libkind[3]

[1] Departamento de Computación, Facultad de Ciencias Exactas y Naturales, Universidad de Buenos Aires, Argentina,
[2] Instituto de Investigación en Ciencias de la Computación (UBA-CONICET) 0+inf, Buenos Aires, Argentina
[3] Centro de Referencia en Levaduras y Tecnología Cervecera (CRELTEC) Instituto Andino Patagónico de Tecnologías Biológicas y Geoambientales (IPATEC, CONICET - UNComahue) Bariloche, Argentina

Abstract. La viabilidad y vitalidad de la levadura Saccharomyces cerevisiae es crucial para la producción cervecera. El método tradicional para evaluar la viabilidad, el recuento manual con azul de metileno, es laborioso y propenso a variabilidad dependiendo el individuo que las cuente. Este trabajo aborda el desarrollo de un algoritmo para el conteo y clasificación automática de levaduras de cerveza a partir de imágenes de microscopio utilizando herramientas de inteligencia artificial, con el objetivo de optimizar la reutilización y el control de calidad del insumo para la industria. El sistema propuesto integra dos etapas principales: la detección de células se realiza mediante la red neuronal YOLO (You Only Look Once), mientras que la clasificación en vivas o muertas se lleva a cabo utilizando una Support Vector Machine (SVM) sobre histogramas del canal de saturación en el espacio de color HSV. Este enfoque combinado ofrece una solución robusta y automatizada para la cuantificación y clasificación de levaduras, reduciendo la variabilidad del análisis manual y optimizando el control de calidad en la producción cervecera. Las metodologías desarrolladas serían incorporadas a la aplicación Microbrew.AR.

Keywords: Viabilidad de Levadura de Cerveza, YOLO, Support Vector Machine (SVM), detección de células, clasificación de levaduras

1 Introducción

Los últimos avances en la evaluación de la viabilidad de la levadura para la elaboración de cerveza han aprovechado los desarrollos en computer vision y las técnicas de aprendizaje automático. Gomide et al., 2021 desarrolló un sistema que utiliza redes neuronales convolucionales para contar automáticamente levaduras viables e inviables, mejorando la precisión sobre los métodos manuales. Feizi et al., 2017 presentó una plataforma portátil de microscopía sin lente que utiliza un support vector machine (SVM) para clasificar las células de levadura

como vivas o muertas, demostrando una precisión comparable a los métodos tradicionales. Saldi et al., 2014 exploró métodos de tinción multi-fluorescentes para evaluar la viabilidad y vitalidad de las levaduras, destacando la importancia de la actividad metabólica en la fermentación. Ohnuki et al., 2014 utilizó el procesamiento de imágenes para analizar los cambios morfológicos de la levadura durante la fermentación, revelando características dinámicas significativas que pueden informar las prácticas de elaboración de cerveza. En conjunto, estos estudios destacan el potencial de las técnicas de imagen innovadoras para mejorar la evaluación de la calidad de la levadura en la elaboración de cerveza, mejorando en última instancia la consistencia del producto y la eficiencia de la fermentación.

La evaluación de la viabilidad y vitalidad de la levadura cervecera es crucial para garantizar la calidad y consistencia de la cerveza. Los métodos tradicionales, como la tinción con azul de metileno, presentan limitaciones, lo que ha llevado a investigar técnicas más precisas (Heggart et al., 2000). La citometría de flujo, que utiliza colorantes fluorescentes como el oxonol, ha demostrado ser prometedora para determinar simultáneamente la viabilidad de la levadura y el recuento de células con un error reducido en comparación con los métodos convencionales (Boyd et al., 2003). La citometría basada en imágenes ha surgido como un enfoque eficaz, utilizando diversas tinciones fluorescentes para medir tanto la viabilidad como la vitalidad (Saldi et al., 2014). Un nuevo método que emplea el colorante fluorescente PO-TEDM-1 y un citómetro de imagen automatizado ha demostrado una gran precisión en la determinación de la viabilidad y el recuento de células de levadura (Atanasova et al., 2019). Estas técnicas avanzadas permiten a las fábricas de cerveza mejorar el control de calidad y las capacidades de seguimiento del proceso, lo que puede dar lugar a resultados de fermentación más uniformes y a una mejor producción de cerveza (Boyd et al., 2003; Saldi et al., 2014).

MicroBrew.AR[4]es una aplicación desarrollada para celulares y tablets que permite hacer más dinámico y sencillo el análisis para la cantidad y calidad de levadura en la producción cervecera, buscando establecer su viabilidad para sucesivas fermentaciones. Es la primera App, desarrollada por el Instituto Andino de Tecnologías Biológicas (IPATEC), orientada a dar soporte de los productores artesanales de cerveza. La misma, permite a los productores calcular la cantidad de levaduras necesarias para un nuevo lote de elaboración, reutilizando de manera eficiente las células vivas presentes en las levaduras recuperadas.

Actualmente, la aplicación cuenta con dos métodos principales para el conteo de levaduras vivas y muertas:

- **Conteo manual**: El productor observa las imágenes obtenidas con un microscopio y registra manualmente la cantidad de células vivas y muertas.
- **Delimitación asistida**: A través de la aplicación, el usuario puede cargar una imagen y delimitar manualmente las áreas donde se encuentran las células vivas o muertas.

[4] https://www.microbrew.com.ar/

En este contexto, este artículo tiene como objetivo el desarrollo de metodologías que busquen optimizar ambas tareas mediante el desarrollo de un sistema de conteo automático de levaduras a partir de imágenes cargadas por el productor en la aplicación. Se busca proporcionar al usuario una herramienta de recuento con la menor intervención para facilitar la utilización de la aplicación.

Este trabajo propone:

- Realizar un conteo automático en las imágenes de microscopio a partir de un modelo de detección profundo basado en YOLOv5.
- Clasificar las levaduras en vivas y muertas a través de histogramas generados en los espacios de color y un discriminador de tipo SVM.

En la sección 2, se describe el dataset empleado para entrenamiento y testeo, además del pre-procesamiento realizado para adaptarlo como entrada a los algoritmos. En la sección 3, se caracterizan los dos modelos utilizados en clasificación y detección, y cómo son entrenados con el dataset a utilizar. Asimismo, en la sección 4, se hace uso de una métrica personalizada para evaluar ambos modelos, y decidir cuál es el óptimo en este caso de uso. Por último, en la sección 5 se exponen los beneficios del algoritmo desarrollado en el ámbito donde es implementado.

2 Dataset

Con el objetivo de desarrollar un modelo de clasificación y detección de células de levaduras, se llevó a cabo un estudio partiendo de un conjunto de 50 imágenes de laboratorio de levaduras de cerveza CRELTEC, cada una acompañada de un archivo en formato .txt que indica la cantidad total de células y cuántas de ellas son células muertas.

Las imágenes se encuentran divididas en dos especies de levaduras distintas, diferenciadas por características morfológicas como el tamaño celular y la densidad de células por imagen. En particular, el dataset incluye 40 imágenes correspondientes a la especie "Saccharomyces cerevisiae - Fermentis S-04" y 10 imágenes de la especie "Saccharomyces eubayanus - CRUB1568". La principal diferencia entre ambas radica en el tamaño celular, siendo las células de "Saccharomyces eubayanus - CRUB1568" más pequeñas y con una mayor densidad por imagen en comparación con "Saccharomyces cerevisiae - Fermentis S-04".

Para la identificación de células muertas, se empleó un pigmento de laboratorio que las tiñe de color azul, permitiendo su diferenciación visual de las células vivas. Asimismo, en el conjunto de datos se identificaron células superpuestas, resultado del proceso de gemación y del estado de movilidad de las mismas.

Es importante destacar que los conteos manuales del dataset se realizaron en una porción específica de cada imagen, la cual corresponde a la sección observada microscópicamente por los productores al momento del conteo mediante la aplicación utilizada.

Cada una de las imágenes fue procesada para identificar las regiones individuales correspondientes a cada célula en la captura de microscopio. Estas regiones, denominadas ROI (Regions of Interest), fueron extraídas y etiquetadas manualmente con el propósito de clasificar cada una según su estado: VIVA o

Imágenes del conjunto de datos

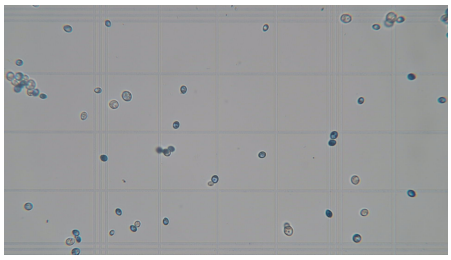


Fig. 1: Saccharomyces cerevisiae

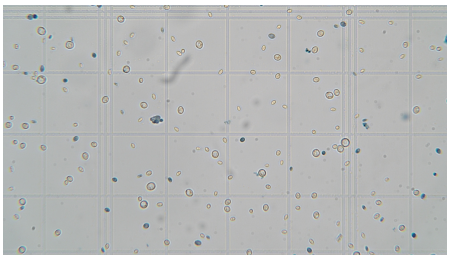


Fig. 2: Saccharomyces eubayanus

MUERTA. Este paso es fundamental para el desarrollo de modelos de clasificación más precisos y robustos. En este proceso fue utilizada una aplicación especialmente desarrollada para este propósito.

Además, se utilizó un segundo dataset (Dataset 2) con imágenes provenientes de un entorno no controlado. Se emplearon imágenes subidas a la aplicación por los usuarios con el fin de construir un dataset más representativo de situaciones reales. Las mismas fueron preprocesadas de la misma manera que las obtenidas en el laboratorio. Esto permitió entrenar y evaluar el modelo en imágenes similares a las que serían proporcionadas a la aplicación en su uso real.

Combinando ambos conjuntos de datos, en el conjunto de *Testeo* se identificaron 3.297 células vivas y 716 células muertas, mientras que en el conjunto *Train* se contabilizaron 12.459 células vivas y 4.531 células muertas.

Dentro del conjunto de *Testeo*, 608 células vivas y 413 células muertas corresponden al *Dataset 1*, mientras que 2.689 células vivas y 303 células muertas pertenecen al *Dataset 2*. En cuanto al conjunto de *Entrenamiento*, 5.345 células vivas y 3.138 células muertas provienen del *Dataset 1*, y 7.114 células vivas junto con 1.393 células muertas provienen del *Dataset 2*.

Los recuentos evidencian un desbalance en los datos para la tarea de clasificación, mientras que no existe tal desbalance en la tarea de detección pues células vivas y células muertas son consideradas dentro de la misma clase, es decir, no habrá distinción entre ellas durante la detección.

2.1 Pre-procesamiento del Dataset

En primer lugar, el pre-procesamiento de las imágenes consiste en el recorte de las mismas al área de interés para el recuento. Para llevar a cabo el recorte, se buscó un promedio del área a analizar en las imágenes originales, mediante la identificación de los bordes de la misma con el algoritmo findpeaks de Scipy.

Debido a que el objetivo es la detección de regiones de interés, se duplicó el dataset combinado de forma que el primero tuviese las imágenes en blanco y negro y el segundo en color original. Esto permitirá comparar eficiencia en la detección de las levaduras. Para la clasificación será obviamente necesario usar el dataset en color pues el color aportará información adicional.

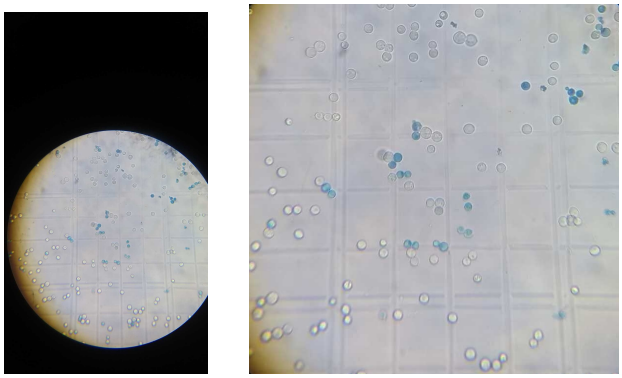
En segundo lugar, se llevó a cabo un proceso de aumento de la cantidad de datos. El mismo consistió en la rotación de las imágenes, tanto aquellas en blanco y negro como aquellas en color, en cuatro direcciones: original, 90° a izquierda, 90° a derecha y 180° a izquierda. De esta manera, se ha logrado aumentar el tamaño original del dataset por cuatro veces.

Finalmente, el dataset se completó al etiquetar manualmente cada una de las imágenes del dataset combinado usando una herramienta especifica de etiquetado desarrollada con este propósito; permitiendo así su utilización directa a la hora de entrenar y evaluar la eficiencia de los modelos al identificar las ROIs y su correspondiente etiqueta: VIVA o MUERTA.

Para el caso de las imágenes obtenidas de la aplicación, existe un paso adicional en el pre-procesamiento. Dado que las imágenes provenían de un entorno no controlado, no solo contenían la región capturada a través del microscopio, sino también diversos artefactos, especialmente en los bordes y el fondo. Por esta razón, fue necesario incorporar un método capaz de identificar y extraer únicamente la zona correspondiente a la imagen del microscopio, eliminando los elementos no deseados.

El modelo implementado para obtener el área de interés, sobre las cuales se distinguirán y contarán las células, sigue una serie de pasos. Primero, se determina si la imagen tiene una forma rectangular. En caso de que así sea, se procede a recortarla, bajo la suposición de que la región relevante para el conteo es cuadrada. Luego, se analiza la cantidad de borde presente en la imagen mediante un método de binarización basado en el umbral de Otsu e identificación de regiones conexas. En este contexto, el término "borde" hace referencia específicamente al borde negro que aparece en las imágenes, como se observa en la Fig. 3 (a).

Finalmente, se recorta la región de interés, obteniendo una imagen sobre la cual tiene sentido realizar el conteo de levaduras y su posterior clasificación, como se muestra en la Fig. 3 (b).



(a) (b)

Fig. 3: Imagen original y resultado del recorte de la zona central.

2.2 Dataset de testeo

Para la evaluación de los algoritmos desarrollados, se utilizó un dataset reservado para el testeo que constaba de un conjunto de 20 imagenes de laboratorio con sus respectivas etiquetas realizadas manualmente, y otras 25 imagenes tomadas directamente de la aplicación Microbrew.ar con sus respectivas etiquetas, también etiquetadas a mano.

Para testear sobre ellas, se emplearon las etiquetas generadas por el modelo de detección YOLO junto con las hechas manualmente. Una comparación entre ambas permite generar una métrica que evalúe que tan bien clasifica las células detectadas. De este modo, el algoritmo de clasificación utiliza la salida del proceso de detección de células, lo que da lugar a un sistema automatizado capaz de abordar ambas tareas de manera integral.

3 Algoritmos

3.1 Modelo de detección

El método empleado para la segmentación de células de levadura se basó en la utilización de un modelo YOLOv5 previamente entrenado en el conjunto de datos COCO (*Common Objects in Context*). Este dataset, de gran tamaño y diversidad de imágenes anotadas, proporciona una base sólida para entrenar modelos de detección y segmentación.

Para adaptar el modelo entrenado a la tarea específica, se realizó un entrenamiento utilizando el dataset 1 en blanco y negro. La hipótesis inicial se basa en que en el reconocimiento de las levaduras para su conteo no depende de los colores, por lo que células muertas y células vivas deberían corresponder ambas a una misma clase, *célula*. Se realizó una comparación de los modelos y era este el que se desempeñaba mejor.

Una vez entrenado, se utilizó el conjunto de imagenes de laboratorio reservadas para testeo para evaluar su rendimiento. Durante esta evaluación, el modelo generó un archivo .txt por cada imagen, indicando las regiones de interés (ROI) detectadas. Para cada ROI, el archivo especifica la clase asignada (en este caso, la única clase es *célula*) y el nivel de confianza asociado a su detección como levadura.

Con el objetivo de minimizar la sobre-estimación en el número de regiones de interés (ROI) identificadas por el modelo, se implementó el método Mean-Shift. Este procedimiento permite eliminar las superposiciones entre ROI que corresponden a una misma región, pero que han sido identificadas más de una vez. Así, cada región de interés es detectada únicamente una vez.

A partir de los niveles de confianza reportados, se generaron diferentes subconjuntos de detecciones aplicando umbrales de confianza específicos: 0.45, 0.5, 0.55, 0.6, 0.65, 0.7, 0.75, 0.8, 0.85 y 0.9. En cada caso, una ROI fue incluida en la nueva detección únicamente si su nivel de confianza superaba el umbral correspondiente. De esta manera, se analizó la eficiencia del modelo en la detección de levaduras bajo distintos umbrales de confianza. A partir del análisis de diferentes métricas, se determinó que las detecciones con un nivel de confianza superior a

0.45 en YOLO deben considerarse válidas e incluirse en el conjunto de células a clasificar.

Detección por YOLO

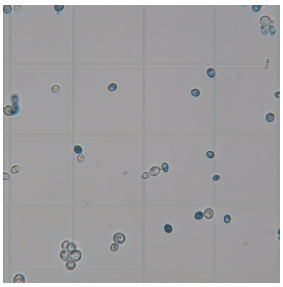


Fig. 4: Original

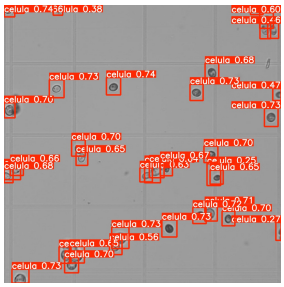


Fig. 5: YOLO

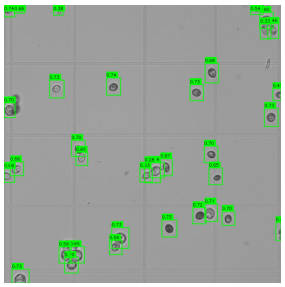


Fig. 6: Mean-Shift

Luego de definir el nivel de confianza, se llevó a cabo un proceso de fine-tuning utilizando el dataset 2. El objetivo de este ajuste fue permitir que el modelo aprendiera las características generales de las células a detectar, tales como el tamaño y la forma, para luego especializarlo y mejorar su capacidad de generalización en imágenes .

El entrenamiento se llevó a cabo de manera progresiva. Se tomó el modelo base y se re-entrenó utilizando un primer subconjunto de imágenes. Luego, el modelo resultante fue nuevamente fine-tuneado con el segundo subconjunto, y así sucesivamente hasta completar el proceso con cinco subconjuntos, obteniendo un total de seis modelos entrenados en diferentes etapas. Este enfoque permitió mejorar la adaptación del modelo a las condiciones de variabilidad presentes en las imágenes aportadas por los usuarios, favoreciendo su desempeño en escenarios menos controlados.

3.2 Modelos de clasificación

El análisis del espectro de color consiste en generar un histograma de color, donde el eje horizontal representa la variedad de colores y el eje vertical indica la cantidad de píxeles correspondientes a cada tono. Las imágenes originales estaban en el espacio de color RGB, por lo que fueron convertidas al espacio de color HSV (Hue, Saturation, Value). Este espacio de color ofrece una ventaja significativa: mientras que en RGB el color, la saturación y el brillo están codificados en los tres canales, en HSV estos se separan en canales independientes. En la Fig. 7 observamos un ejemplo de dos imágenes donde se presentan los 3 canales del HSV. Se observa a simple vista como el canal 'Saturation' es aquel que, en ambas imagenes, permite diferenciar facilmente células vivas de muertas. Por esto, 'Saturation' facilitaría la creación de histogramas de color por ROI que permitiría la la discriminación entre las clases a clasificar, células vivas y muertas.

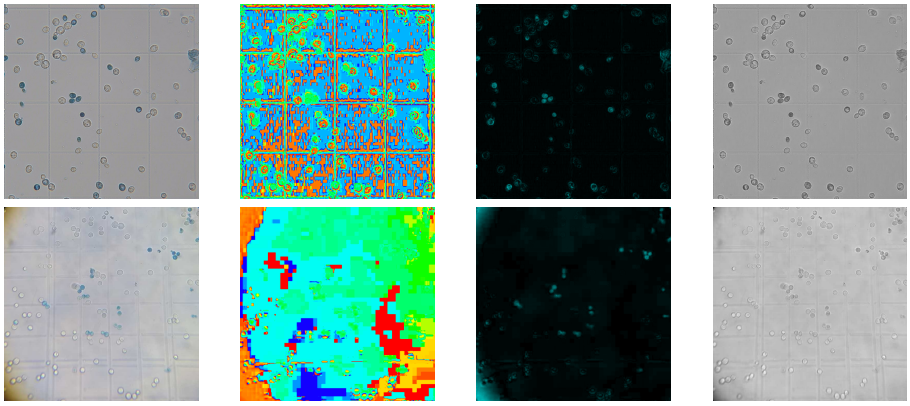


Fig. 7: Imágenes original, con diferenciación de canales Hue, Saturation y Value.

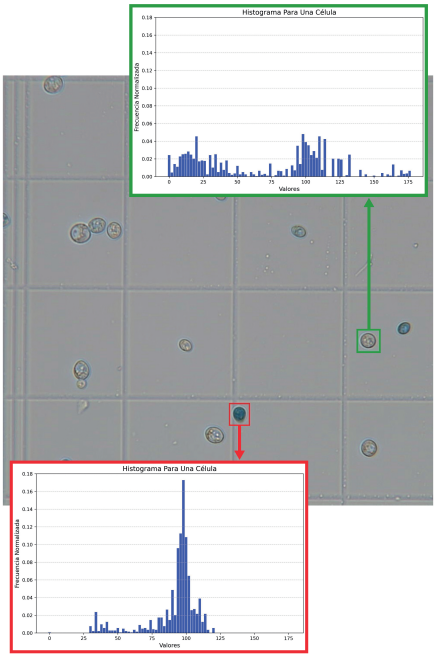
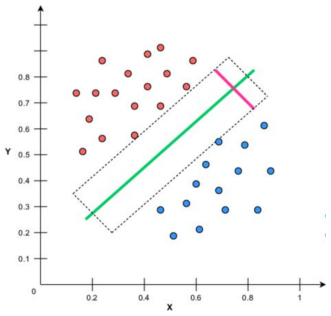


Fig. 8: Imagen sobre cuyas células se realizaran histogramas. La célula marcada en rojo corresponde con una célula muerta mientras que la verde, con una viva.

A modo de ejemplo, en la Fig. 8 se presenta una imagen y el histograma de dos células, una viva y una muerta para observar las diferencias. Los histogramas se construyen acumulando los valores del Hue de los pixeles dentro del cuadro de

detección (RoI) en un histograma de X bins, pero la representación es análoga en el caso de utilizar 'Saturation'. Los histogramas son normalizados para que la suma total sea 1 y tenga naturaleza de probabilidad.

3.2.1 Modelo de clasificación El algoritmo propuesto se basa en el análisis del espectro de color de las imágenes a clasificar, complementado con un método de clasificación supervisado. Durante el desarrollo del modelo final se utilizó *Support Vector Machine* (SVM) con *kernel* 'RBF', utilizando la implementación disponible en la biblioteca `sklearn`.



SVM

Fig. 9: Ejemplo de fronteras discriminantes para el modelo de clasificación utilizado (Ha et al., 2023).

El método *Support Vector Machine* (SVM) es un algoritmo de optimización que permite trazar una frontera de clasificación basada en una frontera óptima calculada en base a los ejemplos de entrenamiento y que busca maximizar el margen de separación (vector rosa en Fig. 9). Si bien el ejemplo muestra un caso de dos poblaciones que son linealmente separables con una recta, el SVM resuelve también casos donde la frontera no es lineal. En estos casos, la utilización de un Kernel especial, como el Radial Basis Function (RBF), incrementa el espacio de proyección permitiendo el trazado del margen de discriminación.

La figura 9 ejemplifica el trazado de la fronteras de clasificación del método de clasificación binaria utilizado para diferenciar las levaduras vivas de las muertas.

3.2.2 Entrenamiento de los modelos de clasificación El entrenamiento del clasificador se realiza utilizando ROIs extraídas de las imágenes del conjunto de entrenamiento. Para ello, cada ROI, inicialmente en el espacio de color RGB, se convierte al espacio HSV y se extrae el canal deseado. A partir de este canal, se genera un histograma normalizado de color que se utiliza como entrada para entrenar el clasificador. Se entrenaron dos modelos de clasificación, donde uno utiliza el canal 'Hue' para hacer las clasificaciones, y el otro, el canal 'Saturation'.

Se llevó a cabo una búsqueda exhaustiva de hiperparámetros (*grid search*) para optimizar la precisión del clasificador y para evaluar la eficacia de los modelos, se utilizó el conjunto de entrenamiento mediante validación cruzada (*cross-validation*), dividiendo aleatoriamente las imágenes.

Una vez seleccionado el mejor modelo, este se entrena con el conjunto de datos definitivo y se guarda para su posterior uso. Luego, se emplea para clasificar las imágenes del conjunto de validación, evaluando así su desempeño real.

La clasificación de imágenes no utilizadas durante el entrenamiento, como las de validación, requiere un paso previo de detección de las células. Se emplea el modelo descrito en el informe inicial, basado en YOLO, para identificar las regiones de interés (ROIs) dentro de las imágenes. Este modelo retorna las coordenadas de dichas regiones, que son luego utilizadas para la clasificación.

Cada región de interés es recortada de la imagen original y procesada individualmente. Como estas regiones están en el espacio de color original (RGB), primero son convertidas al espacio HSV, donde se toma únicamente el canal deseado para generar el histograma de color. Este histograma se utiliza como entrada para el clasificador, que determina si la célula pertenece a la clase de células vivas o muertas. Este procedimiento se repite para cada ROI de la imagen, obteniendo como resultado una categorización de todas las células detectadas.

4 Resultados

4.1 Modelos de detección

En esta sección se presentan los resultados de la detección de células utilizando el algoritmo YOLO. Como se mencionó en la sección 3.1, para seleccionar el mejor modelo basado en YOLO, se filtró a las regiones de interés detectadas mediante sus niveles de confianza, con el objetivo de encontrar el umbral mínimo mediante el cual determinadas métricas se maximizan.

- **Falsos Negativos:** Un falso negativo ocurre cuando una instancia positiva en los datos no es correctamente identificada por el modelo o algoritmo de detección. En este caso práctico, representa la cantidad de células que el modelo no detectó, que sí fueron identificadas por el conteo manual.
- **Falsos Positivos:** Un falso positivo ocurre cuando una instancia negativa en los datos es incorrectamente clasificada como positiva por el modelo, generando sobre-detección. En este caso práctico, representa aquellas instancias donde el modelo detectó una célula donde no hay una.

Para el análisis de los falsos positivos y falsos negativos, se representaron gráficamente estos parámetros en función del umbral de confianza empleado. Adicionalmente, se incluyó un gráfico de la precisión de cada umbral. Los resultados para cada umbral provienen del promedio de cada métrica, proveniente de las 20 imágenes de laboratorio reservadas para el testeo.

Se observa en el gráfico que la curva de *accuracy* (calculada como cantidad de detecciones sobre cantidad de células detectadas totales) desciende hasta valores cerca del 0 bajo niveles de confianza altos. Lo mismo es consistente con el hecho que a medida que el nivel sube, la cantidad de células detectadas es menor.

Asimismo, al analizar los falsos positivos y falsos negativos, se concluye que el nivel de confianza óptimo se encuentra en 0,5 para imágenes a color y en 0,55 para imágenes en blanco y negro. A partir de estos puntos, se observa un incremento en la proporción de falsos negativos, y una proporción del 35% de falsos positivos, además de que la accuracy se encuentra en un valor alto en ambos casos. Sin embargo, se tomará como umbral de nivel de confianza, en ambos casos, 0.45 pues la mejora no es sustancial y se prefiere no disminuir el valor del accuracy.

Comparación de resultados entre detección manual y automática (YOLO)

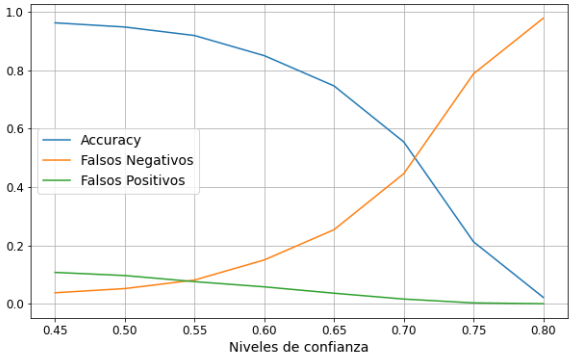


Fig. 10: Imágenes de laboratorio en blanco y negro

Se presentan en Tabla 1 los resultados de las métricas para los dos datasets, donde la métrica usada es la siguiente:

$$\text{Métrica} = \frac{\#\text{ células detectadas correctamente}}{\#\text{ células detectadas manualmente o células totales}}$$

Dataset	Métrica - Detección
1	1.0
2	0.984
1 y 2	0.991

Table 1: Estadísticas - Modelo de detección

A continuación, se presenta una tabla que muestra cómo mejora la detección al realizar un proceso de fine-tuning sobre las imágenes del segundo dataset. Los resultados obtenidos muestran una mejora significativa en la métrica a medida que el modelo es reentrenado con nuevas imágenes. Especialmente en la segunda y tercera etapa, se observa un incremento en el rendimiento.

Etapa	Modelo de partida	Dataset utilizado	Métrica
1	Modelo base	Imágenes de laboratorio	0.647
2	Modelo fine-tuneado 1	Primer grupo de imágenes	0.845
3	Modelo fine-tuneado 2	Segundo grupo de imágenes	0.907
4	Modelo fine-tuneado 3	Tercer grupo de imágenes	0.905
5	Modelo fine-tuneado 4	Cuarto grupo de imágenes	0.905
6	Modelo fine-tuneado 4	Quinto grupo de imágenes	0.984

Table 2: Esquema del proceso de fine-tuning por etapas con la métrica obtenida en cada fase para imágenes del segundo dataset

4.2 Modelos de clasificación

En esta sección se presentan los resultados de la clasificación de células mediante el modelo descrito en la Sección 3.2: Máquinas de Vectores de Soporte (SVM), aplicados a las imágenes de testeo. Este modelo fue entrenado con las etiquetas manuales de cada ROI en las imágenes de entrenamiento en color del dataset 1 y 2. Para el análisis de rendimiento, se consideraron únicamente aquellas regiones identificadas por el algoritmo YOLO que coincidieron con las regiones delimitadas manualmente.

En la Tabla 3 se muestra la eficiencia del modelo en los diferentes datasets. Además, se compara la perforance del SVM con otro clasificador del estado del arte denominado Ranform Forest (RF). Sea A el conjunto de células detectadas correctamente por el modelo de detección, la métrica usada es la siguiente:

$$\text{Métrica} = \frac{\text{cantidad de células clasificadas correctamente pertenecientes a A}}{\#\ \text{A}}$$

Dataset	Hue		Saturation	
	SVM	RF	SVM	RF
1	0.897	0.425	0.896	0.590
2	0.891	0.866	0.927	0.871
1 y 2	0.893	0.670	0.913	0.746

Table 3: Estadísticas - Modelo de clasificación SVM y RF

Se observa que en el dataset 2 la clasificación presenta una mejor perfomance que en las imágenes de laboratorio. Esto puede deberse a que el modelo fue entrenado para ajustarse a situaciones reales. Además se observa que, para todos los casos, el modelo SVM performa mejor que un Random Forest.

En la figura 11, se exponen las matrices de confusión de ambos modelos, realizadas a partir de todas las células presentes en el dataset de testeo. Se observa que la performance del modelo con canal 'Saturation' supera a la del modelo

'Hue', además de que la cantidad de falsos negativos se reduce considerablemente.

En conclusión, el modelo que se utiliza en el pipeline del algoritmo es el modelo SVM que selecciona el canal 'Saturation' para hacer sus clasificaciones, en vista de lo observado por las métricas y las matrices.

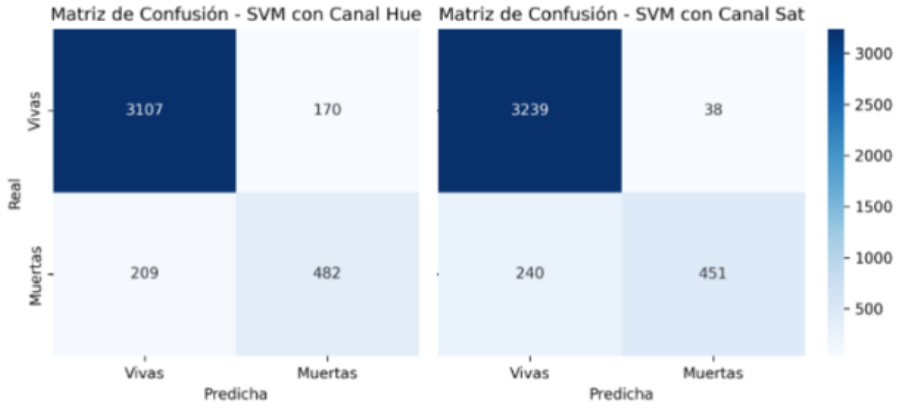


Fig. 11: Matrices de confusión para modelos con canal Hue y Saturation, respectivamente

5 Discusión y conclusiones

El sistema desarrollado, que combina un modelo de detección basado en YOLO y un clasificador SVM, permite abordar de manera eficiente la automatización del conteo y clasificación de levaduras de cerveza en imágenes de microscopio.

Los resultados obtenidos muestran un poder de detección cercano a 0.98, lo que evidencia una adecuada capacidad de generalización del modelo, incluso en imágenes provenientes de entornos no controlados. A su vez, la métrica alcanzada en el proceso de clasificación representa un avance significativo, acelerando considerablemente el proceso de análisis de viabilidad de las levaduras.

La integración del sistema a la aplicación Microbrew.AR supone una mejora sustancial respecto a las metodologías implementadas hasta el momento, reduciendo tanto el tiempo necesario para el conteo como el margen de error asociado al factor humano. La combinación de detección automatizada con una supervisión rápida por parte del usuario final permitiría optimizar el análisis de viabilidad celular, contribuyendo directamente a la mejora en la calidad de las fermentaciones y a una reutilización más eficiente de los insumos.

Este trabajo fue financiado por el proyecto UBACyT 20020220400306BA, y el Proyecto Federal de Innovación RN-2-PFI 2023 (Convenio Nro. 29265373/2018).

References

Atanasova, M., Yordanova, G., Nenkova, R., Ivanov, Y., Godjevargova, T., & Dinev, D. (2019). Brewing yeast viability measured using a novel fluorescent dye and image cytometer. *Biotechnology & Biotechnological Equipment*, *33*(1), 548–558.

Boyd, A. R., Gunasekera, T. S., Attfield, P. V., Simic, K., Vincent, S. F., & Veal, D. A. (2003). A flow-cytometric method for determination of yeast viability and cell number in a brewery. *FEMS yeast research*, *3*(1), 11–16.

Feizi, A., Zhang, Y., Greenbaum, A., Guziak, A., Luong, M., Chan, R. Y. L., Berg, B., Ozkan, H., Luo, W., Wu, M., et al. (2017). Yeast viability and concentration analysis using lens-free computational microscopy and machine learning. *Optics and Biophotonics in Low-Resource Settings III*, *10055*, 32–38.

Gomide, J. V. B., Cunha, E. V., & Gomide, G. B. (2021). Automatic yeast detection and counting using computer vision techniques. *Workshop de Visão Computacional (WVC)*, 25–30.

Ha, K., Naseem, U., Keya, S., Ashfia Maitra, Mithu, K., & Alam, M. G. R. (2023). A systematic review on airlines industries based on sentiment analysis and topic modeling. *Research Journal*.

Heggart, H., Margaritis, A., Stewart, R., Pilkington, M., Sobezak, J., & Russell, I. (2000). Measurement of brewing yeast viability and vitality: A review of methods. *Technical quarterly-Master Brewers Association of the Americas*, *37*(4), 409–430.

Ohnuki, S., Enomoto, K., Yoshimoto, H., & Ohya, Y. (2014). Dynamic changes in brewing yeast cells in culture revealed by statistical analyses of yeast morphological data. *Journal of bioscience and bioengineering*, *117*(3), 278–284.

Saldi, S., Driscoll, D., Kuksin, D., & Chan, L. L.-Y. (2014). Image-based cytometric analysis of fluorescent viability and vitality staining methods for ale and lager fermentation yeast. *Journal of the American Society of Brewing Chemists*, *72*(4), 253–260.