

Modelado II

Adrián Matute, Pablo Martínez, Osvaldo Del Valle, Andres Callealta, Jorge Martínez
Tecnológico de Monterrey Campus Querétaro, México

1. Técnica de modelado

En esta fase de modelado continuamos utilizando la arquitectura YOLOv8, seleccionada previamente por su capacidad para equilibrar precisión, velocidad y adaptabilidad en escenarios con variaciones de iluminación, como las imágenes diurnas y nocturnas. Como se explicó en el reporte anterior, esta elección se basa en un análisis detallado de su rendimiento frente a otras arquitecturas como Faster R-CNN.

YOLOv8 hereda las mejoras de versiones anteriores, incluyendo optimizaciones en su backbone, basado en CSPNet, y en su neck, que utiliza PANet para integrar información de múltiples escalas. Además, su diseño anchor-free y su avanzado sistema de data augmentation contribuyen a su eficiencia en condiciones desafiantes y entornos con recursos limitados.

Este reporte se enfoca en la mejora del modelo por medio del entrenamiento con el dataset original separado en conjuntos de día y noche, con la intención de crear dos modelos los cuales se puedan usar dinámicamente para realizar inferencias dependiendo de las características ambientales.

2. Supuestos de modelado

Se espera que la generación de modelos distintos para el día y noche mejore la precisión en comparación con las inferencias que realiza el modelo general. Nos basamos en el supuesto de que la variabilidad entre el día y la noche están afectando la capacidad del modelo de visión y aumentando el ruido. Al generar los dos modelos podremos seleccionar cual usar por medio de las características ambientales.

3. Diseño de pruebas

La detección de objetos por medio de múltiples bounding boxes presenta 3 resultados que utilizaremos para medir la precisión de la detección.

- Las cajas predichas, coordinadas del centro de la caja, ancho y alto.
- Las clases a las que pertenece cada objeto para dichas cajas, en nuestro caso es

únicamente *vaca* o *fondo*.

- La confianza que tiene el modelo de que la clase predicha.

3.1 Métricas de evaluación

Para evaluar la calidad y validez del modelo en la tarea de detección y conteo de vacas en la fila de espera, se utilizaron las siguientes métricas clave:

- **Precisión:**
 - a. La proporción de verdaderos positivos entre todas las predicciones de clases, nos muestra que tan propenso es el modelo a generar falsos positivos.
 - b. Resultado esperado: Superior a 85% para garantizar un conteo confiable en imágenes diurnas y nocturnas.
- **Recall (Sensibilidad):**
 - a. La proporción de verdaderos positivos sobre los positivos esperados, mostrando la habilidad del modelo de predecir todos los objetos presentados.
 - b. Resultado esperado: Superior al 90%, para minimizar vacas no detectadas.
- **mAP@50 y mAP@50-95:**
 - a. mAP@50: Precisión media para un umbral de intersección sobre unión (IoU) del 50%.
 - b. mAP@50-95: Precisión media en múltiples umbrales de IoU (50%-95%).
 - c. Resultado esperado:
 - i. mAP@50: Superior al 95% en condiciones diurnas.
 - ii. mAP@50-95: Superior al 80% para condiciones mixtas.

Estas métricas permiten calcular un porcentaje de error y observar cómo el modelo se ajusta a los objetivos planteados, tanto en términos de detección como de conteo.

3.2 División del Dataset

El dataset total de 8,087 imágenes fue dividido cuidadosamente para asegurar una representación balanceada de condiciones diurnas y nocturnas:

1. Porcentaje de división:
 - 80% para entrenamiento: El modelo ajusta

sus parámetros para mejorar la detección de vacas.

- 20% para validación: Evalúa el rendimiento del modelo en imágenes no vistas previamente.
2. Subconjuntos por iluminación:
- Imágenes diurnas: Capturadas entre las 6:00 am y las 8:00 pm.
 - Imágenes nocturnas: Capturadas entre las 8:01 pm y las 5:59 am.
 - Cada modelo fue entrenado y validado en su respectivo subconjunto, mejorando la precisión en cada condición de iluminación.

3.3 Plan de entrenamiento, pruebas y evaluación

- **Entrenamiento:**
 - Modelo base: Se utilizó Transfer Learning con un modelo pre entrenado en el dataset COCO, para aprovechar las características generales aprendidas.
 - Épocas: Se realizaron hasta 150 iteraciones
- **Validación y pruebas:**
 - El conjunto de validación fue utilizado para calcular las métricas de precisión, recall y mAP en imágenes no vistas previamente.
 - Se generaron gráficos de pérdida y métricas para observar la estabilidad y el rendimiento del modelo durante las épocas de entrenamiento.
- **Optimización y ajustes:**
 - Tasa de aprendizaje: Ajustada iterativamente para garantizar una convergencia eficiente.
 - Tamaño de lote: Modificado según los recursos de hardware disponibles.
 - Parada temprana: Implementada para detener el entrenamiento si no se observaron mejoras significativas en las métricas de validación.

3.4 Resultados Esperados y Actividades de Prueba

1. Conteo de vacas en la fila de espera:
- Prueba: Evaluar la precisión del modelo para identificar la cantidad exacta de vacas en imágenes.
 - Condiciones: Considerar situaciones con vacas solapadas o en movimiento.
 - Métricas clave: Precisión y recall superiores al 85%.

2. Análisis de aglomeraciones para identificar cuellos de botella:

- Prueba: Detectar patrones de aglomeración en secuencias de imágenes y evaluar cambios en la densidad de vacas en el área de espera.
- Evaluación: Se utilizaron métricas de recall en series temporales para determinar momentos de alta congestión.

3.5 Justificación del Diseño de Test

El diseño de test asegura que las métricas elegidas sean representativas de los objetivos planteados:

- Precisión y Recall: Directamente alineadas con la capacidad del modelo para cumplir con el conteo de vacas.
- mAP: Garantiza que las detecciones sean consistentes a través de múltiples umbrales, lo que es crucial para minimizar errores en el análisis de aglomeraciones.

Estas pruebas validan que el modelo cumple con los requisitos técnicos y de negocio, maximizando su capacidad para detectar patrones útiles para la toma de decisiones en el sistema de ordeño.

4. Ajuste de parámetros

Para ambos modelos, diurno y nocturno, se consideraron y ajustaron los siguientes parámetros de entrenamiento.

- **Épocas**
 - Representan la cantidad de veces que el modelo entrena con todo el dataset.
- **Tamaño de lote**
 - La cantidad de ejemplos que se corren en paralelo para calcular la pérdida.
- **Weight decay**
 - Técnica de regularización que castiga la alta complejidad del modelo agregando a la pérdida el cuadrado del valor de los pesos multiplicados por el coeficiente de weight decay.
- **Taza de aprendizaje**
 - Proporción por la cual cambiar los pesos en base a la pérdida.

5. Descripción del modelo

5.1 Modelo diurno

El modelo fue entrenado con el 80% del dataset de día (3,745 imágenes), y el 20% fue usado para la validación (936 imágenes).

- **Épocas:** 100
 - Se dejó con el valor por defecto ya que en el entrenamiento se guardaría siempre la mejor versión del modelo. Más adelante no se notó una mejoría considerable con más épocas.
- **Tamaño de lote:** 16
 - Se dejó con el valor por defecto ya que tamaños más grandes caían en problemas de aumentar mucho el tiempo entrenamiento por época.
- **Weight decay:** 0.0005
 - Se dejó con este valor ya que utilizar valores muy grandes, cercanos a 1, puede inflar el tamaño de la pérdida y disminuir en exceso el valor de los pesos.
- **Taza de aprendizaje:** 0.01
 - Se dejó con este valor ya que valores muy grandes impiden que el modelo converja en un mínimo local y valores muy pequeños reducen el ajuste haciendo que el modelo se entrene más lento.

5.2 Modelo nocturno

Este modelo fue entrenado con la misma proporción para entrenamiento y validación, 80-20. 2,725 imágenes en pruebas y 681 para validación.

- **Épocas:** 150
 - Se aumentó la cantidad de épocas con la intención de mejorar la detección en la condiciones de la noche.
- **Tamaño de lote:** 64
 - Se aumentaron los ejemplos entrenados en paralelo para compensar el aumento en las épocas y disminución de la taza de aprendizaje.
- **Weight decay:** 0.0001
 - Se disminuyó la influencia de los pesos en el ajuste del modelo para aumentar la complejidad a la que puede llegar el modelo.
- **Taza de aprendizaje:** 0.0002
 - Se disminuyó para tener un ajuste más pequeño en el modelo por lote. Esto con la

intención de detallar el cambio en los pesos para un problema más complejo.

6. Evaluación del modelo

6.1 Modelo diurno

Métrica	Valor
Precisión	96%
Recall	96%
mAP@50	98%
mAP@50-95	87%

6.1.1 Robustez

El modelo presentó un un desenvolvimiento en la condición del día, no obstante, tiene errores detección de objetos, pero su precisión aumentó en 1% con respecto al primer modelo.

6.1.2 Limitaciones

- El modelo llega a encadenar diferentes partes de la vaca, generando súper vacas.
- El modelo algunas veces predice que el ser humano es una vaca, y en la noche hay momentos que no puede identificar si es una vaca.

6.1.3 Dificultades

- **Tiempo de entrenamiento:** El modelo se tardó en entrenar 2.42 hrs por 100 épocas en una GPU 4050.
- **Generalización:** Al entrenar este modelo en condiciones del día, solo podría trabajar en estas condiciones, además al ser muy volátil la iluminación por hora, dificulta que el modelo prediga de manera correcta, ya que la iluminación del día cambia por estación estación del año.

6.2 Modelo nocturno

Métrica	Valor
Precisión	92%

Recall	92%
mAP@50	95%
mAP@50-95	71%

6.2.1 Robustez

El modelo demostró ser menos robusto que el de día, aunque sigue teniendo buenas métricas. Ese bajo rendimiento se puede explicar por la cualidad de los datos, es muy probable que al hacer la bounding-boxes manualmente, hayas errores, sobre todo cuando hay muchas vacas (y por lo tanto mucha intersección), eso se refleja en la métrica (mAP@50-95) de 73%; mostrando un mal ajuste a la intersección con las cajas.

6.2.2 Limitaciones

- El modelo algunas veces predice que no hay vacas en la fila.
- El error cuando se interceptan las bounding boxes de un 50% a un 95% es del 29%.
- La precisión disminuyó en un 3% con respecto al primer modelo, obteniendo un error al contar vacas en la fila y detectarlas del 8%.

6.2.3 Dificultades

- **Tiempo de entrenamiento:** El modelo se tardó en entrenar 43min por 150 épocas en una GPU A100.
- **Dataset:** El dataset de noche cuenta con 3,406 fotos para entrenar el modelo, por lo que, no son suficientes para que el modelo pueda entrenar de la mejor manera debido a la iluminación.

6.3 Comparación con el Modelo Híbrido

El modelo híbrido, entrenado con imágenes de día y de noche, tiene una precisión de 95.14%. A continuación se muestra un cálculo comparativo que evalúa su desempeño en comparación con los modelos especializados de día y de noche:

1. **Modelo Diurno (4681 imágenes):** 96.4% de precisión.
2. **Modelo Nocturno (3406 imágenes):** 92.9% de precisión.
3. **Modelo Híbrido (8087 imágenes):** 95.14% de precisión.

6.4 Conclusión de la Evaluación

Cada modelo presenta ventajas específicas:

- **Modelo Diurno:** Mejor precisión y robustez en condiciones de luz natural. Ideal para monitorear el flujo durante el día y capturar detalles específicos en un ambiente bien iluminado.
- **Modelo Nocturno:** Adecuado para condiciones de poca luz, aunque con limitaciones en cuanto a precisión en alta densidad. Útil para supervisión básica de aglomeraciones durante la noche.
- **Modelo Híbrido:** Ofrece un balance entre precisión y flexibilidad, permitiendo su aplicación en entornos mixtos sin ajustes adicionales.

7. Referencias

Ultralytics. (2024, 1 octubre). YOLO

Performance Metrics. Ultralytics YOLO Docs.

<https://docs.ultralytics.com/guides/yolo-performance-metrics/#object-detection-metrics>