

Marcos Arquitectónicos para la Estimación de Dimensiones de Salmones en 3D: Un Análisis Comparativo de la Estimación de Pose y la Segmentación de Instancia con Visión Estéreo

Sección 1: Metodologías Fundamentales para la Biometría Animal

Para abordar la tarea de estimar las dimensiones de los salmones, es imperativo primero establecer una comprensión teórica clara de las dos metodologías de visión por computadora más pertinentes: la estimación de pose y la segmentación de instancia. Estas técnicas representan enfoques fundamentalmente diferentes para la extracción de información morfológica de imágenes y videos. La elección entre ellas no es trivial, ya que dicta la naturaleza de los datos extraídos, la complejidad del procesamiento posterior y, en última instancia, la precisión y versatilidad del sistema de medición final. Esta sección desglosa cada metodología, define sus aplicaciones en el análisis animal y sienta las bases para una recomendación arquitectónica informada.

1.1 Estimación de Pose Animal (APE): Localización de Puntos de Referencia Anatómicos

La estimación de pose animal (Animal Pose Estimation, APE) es una tarea de visión por computadora que consiste en identificar y rastrear las ubicaciones espaciales de partes clave del cuerpo y articulaciones de un animal a partir de imágenes o videos.¹ El resultado es un conjunto de coordenadas que forman un "modelo cinematográfico" o "modelo basado en esqueleto", el cual representa la postura y la estructura del animal de manera abstracta pero cuantitativa.² Aunque muchas de las técnicas de APE se han adaptado de la estimación de pose humana, su aplicación a animales presenta desafíos únicos debido a la diversidad de morfologías y la falta de grandes conjuntos de datos anotados.³

Aplicación al Salmón

En el caso de un salmón, los "puntos clave" (keypoints) no corresponden a articulaciones tradicionales como en los mamíferos, sino a puntos de referencia morfológicamente significativos. La definición precisa de estos puntos es crucial para la tarea de medición. Un conjunto de puntos clave bien definido para un salmón podría incluir:

- La punta del hocico.
- El centro del ojo.
- El punto de inserción de la aleta dorsal.
- El pedúnculo caudal (la base de la cola).
- Las puntas superior e inferior de la aleta caudal (horquilla).

Este enfoque se utiliza explícitamente en la investigación pesquera y la acuicultura para medir características estandarizadas como la longitud de la cabeza (distancia del hocico al opérculo) y la longitud de horquilla (distancia del hocico a la bifurcación de la cola).⁵ La estimación de estas longitudes se convierte en una tarea de calcular la distancia euclíadiana entre los puntos clave detectados.

Implementación Técnica

Modelos de aprendizaje profundo como DeepLabCut han sido pioneros en la APE para una variedad de especies en entornos de laboratorio.³ Sin embargo, para un sistema de despliegue en tiempo real como el que se está desarrollando, arquitecturas más modernas y eficientes son preferibles. El modelo **YOLOv8-Pose**, parte del ecosistema Ultralytics, es una opción muy relevante. Extiende la arquitectura de detección de objetos YOLOv8 para predecir simultáneamente cuadros delimitadores (bounding boxes) y un conjunto de puntos clave predefinidos para cada objeto detectado.⁹ Este modelo ha sido aplicado con éxito para la estimación de la longitud de peces, demostrando su viabilidad para esta tarea específica.⁷ El resultado del modelo para cada salmón detectado sería un vector de coordenadas 2D (u,v) para cada uno de los puntos clave definidos.

Desafíos Inherentes

La aplicación de APE a salmones no está exenta de dificultades. Los peces son animales de cuerpo blando con una alta capacidad de deformación, lo que significa que su postura puede variar drásticamente, complicando la localización consistente de los puntos clave.⁴ Además, la escasez de datos de entrenamiento etiquetados para especies específicas de peces es un obstáculo significativo, a menudo requiriendo la creación de un conjunto de datos personalizado.³ Finalmente, en un entorno natural o de acuicultura, las occlusiones (cuando un

pez es parcial o totalmente cubierto por otro o por un objeto) pueden hacer que la detección de puntos clave críticos, como la cabeza o la cola, falle por completo, invalidando la medición.⁸

1.2 Segmentación de Instancia: Delineando la Forma Completa

La segmentación de instancia representa un avance significativo sobre otras tareas de visión por computadora como la detección de objetos y la segmentación semántica. Mientras que la detección de objetos dibuja un cuadro delimitador alrededor de un objeto y la segmentación semántica asigna una etiqueta de clase a cada píxel (por ejemplo, "pez" vs. "agua"), la segmentación de instancia logra lo mejor de ambos mundos: detecta cada objeto individual y, además, genera una máscara a nivel de píxel que delineea su forma exacta.¹⁴ Esta capacidad de distinguir entre instancias individuales de la misma clase es fundamental para analizar salmones en un banco, donde los individuos pueden estar muy juntos o superpuestos.¹⁷

El Valor del Resultado

El producto principal de un modelo de segmentación de instancia es una máscara binaria para cada salmón detectado. Esta máscara es una fuente de datos increíblemente rica. A diferencia del conjunto disperso de puntos de la estimación de pose, la máscara contiene información completa y densa sobre la forma 2D del objeto, su contorno preciso, su área total y su distribución espacial.¹⁹ Esta representación literal de la forma es especialmente valiosa para medir dimensiones no lineales, como la longitud a lo largo de un cuerpo curvado, o para estimar la biomasa a partir del área proyectada.²⁰

Implementación Técnica

Arquitecturas canónicas como Mask R-CNN han establecido el estándar de precisión en la segmentación de instancia.¹⁵ Estos modelos suelen operar en dos etapas: primero proponen regiones de interés y luego, para cada región, clasifican el objeto y generan una máscara.¹⁴ Sin embargo, para aplicaciones que requieren alta velocidad, los modelos de una sola etapa son preferibles. El modelo **YOLOv8-Seg**, que el usuario ya está considerando, es una implementación de última generación que integra la segmentación de instancia en el rápido y eficiente marco de YOLOv8.²³ Al igual que YOLOv8-Pose, añade una "cabeza" de predicción adicional a la arquitectura base, pero en este caso, esta cabeza está diseñada para generar las máscaras de segmentación a nivel de píxel para cada objeto detectado.²⁵

Aplicación al Salmón

Para un animal flexible y a menudo curvado como un salmón, la máscara de segmentación captura la verdadera silueta del cuerpo en un instante de tiempo, algo que un conjunto disperso de puntos clave no puede representar completamente.²⁸ Si un salmón está nadando y su cuerpo forma una curva en 'S', la distancia en línea recta entre la cabeza y la cola subestimará drásticamente su longitud real. La máscara, sin embargo, contiene toda la información necesaria para trazar y medir la línea central curvada del pez, proporcionando una base para una medición mucho más precisa.

La distinción fundamental entre estas dos metodologías reside en la naturaleza de la representación de datos que cada una genera. La estimación de pose produce una **representación dispersa y abstracta** del objeto: un esqueleto definido por un número limitado de puntos. Esta representación es computacionalmente ligera y excelente para mediciones predefinidas y lineales entre puntos de referencia anatómicos. Por otro lado, la segmentación de instancia ofrece una **representación densa y literal**: la forma exacta del objeto a nivel de píxel. Aunque computacionalmente más intensiva, esta representación holística es inherentemente una fuente de datos más fiel para la medición precisa de un cuerpo no rígido y curvado como el de un salmón. Esta dicotomía es el eje central sobre el cual debe construirse la arquitectura del sistema de medición.

Sección 2: Enfoques Estratégicos para la Medición de Dimensiones a partir de la Visión 2D

Una vez que se han establecido las metodologías fundamentales, el siguiente paso es analizar en profundidad cómo se pueden extraer dimensiones físicas a partir de sus respectivos resultados. Este proceso no es trivial y revela complejidades ocultas que influyen directamente en la precisión y robustez del sistema final. Esta sección compara directamente las vías de medición que se derivan de los puntos clave y de las máscaras de segmentación, exponiendo las ventajas y desventajas de cada una.

2.1 Medición a través de Esqueletos de Puntos Clave

La utilización de puntos clave para la medición se basa en la geometría de coordenadas. El modelo de estimación de pose proporciona un conjunto de puntos (u, v) en el espacio de la imagen, y las dimensiones se calculan a partir de las distancias entre estos puntos.

Medición Lineal Directa

El método más simple consiste en calcular la distancia euclíadiana entre dos puntos clave de interés. Por ejemplo, para medir la longitud de horquilla de un salmón, se identificarían los puntos clave correspondientes a la punta del hocico (u_h, v_h) y a la bifurcación de la cola (u_t, v_t). La longitud en píxeles L_p se calcularía como:

$$L_p = \sqrt{(u_h - u_t)^2 + (v_h - v_t)^2}$$

Este enfoque es rápido y computacionalmente eficiente, pero solo proporciona una medida de la cuerda (la distancia en línea recta) entre los dos puntos. Si el cuerpo del pez está curvado, esta medida subestimará significativamente la longitud real del contorno.⁶

Aproximación de la Longitud Curva

Para obtener una estimación más precisa de la longitud de un salmón curvado, se puede definir una serie de puntos clave a lo largo de su línea lateral o espina dorsal. Por ejemplo, se podrían definir cinco puntos: P1 (hocico), P2 (detrás del opérculo), P3 (inicio de la aleta dorsal), P4 (pedúnculo caudal) y P5 (horquilla de la cola).

La longitud total curva L_c se aproximaría entonces como la suma de las distancias lineales entre los puntos clave consecutivos:

$$L_c = \sum_{i=1}^{n-1} \text{distancia}(P_i, P_{i+1})$$

donde n es el número de puntos clave. La precisión de esta aproximación es directamente proporcional al número de puntos clave definidos a lo largo de la curva.⁵ Un mayor número de puntos clave puede modelar la curvatura con mayor fidelidad.

Limitaciones del Enfoque de Puntos Clave

A pesar de su aparente simplicidad, este método tiene limitaciones fundamentales. En primer lugar, sigue siendo una **aproximación**. No captura la curvatura real del cuerpo del pez entre los puntos clave. Para lograr una alta fidelidad, se requeriría un número muy elevado de puntos, lo que aumenta drásticamente la complejidad de la anotación manual del conjunto de datos y la dificultad de la tarea de predicción para el modelo.⁵ En segundo lugar, el método es muy frágil. La oclusión o la detección incorrecta de un solo punto clave en la secuencia puede invalidar toda la medición de la longitud. Si el punto P3 es oculto, la cadena se rompe y la longitud total no se puede calcular correctamente.

2.2 Medición a través de Máscaras de Segmentación y Análisis Morfológico: Un Proceso de Múltiples Etapas

Una máscara de segmentación en bruto proporciona la forma completa del salmón, pero no ofrece una medida directa de "longitud" para un objeto curvado. Para extraer esta medida, se requiere un sofisticado proceso de post-procesamiento de imágenes, que se puede

conceptualizar como un pipeline de cuatro etapas. Este pipeline transforma la representación densa de la máscara en una medida escalar precisa de la longitud.

Etapa 1: Refinamiento de la Máscara (Pre-procesamiento)

Las máscaras generadas por las redes neuronales, aunque generalmente precisas, pueden contener imperfecciones como pequeños agujeros, ruido de "sal y pimienta" o bordes irregulares.²⁸ Antes de cualquier medición, es crucial limpiar y refinar estas máscaras para asegurar que los algoritmos posteriores operen sobre una forma limpia y contigua. Las **Operaciones Morfológicas** de OpenCV son la herramienta estándar para esta tarea.³³

- **Apertura (Opening):** Se aplica una operación de erosión seguida de una de dilatación. Este proceso es eficaz para eliminar pequeños artefactos de ruido (píxeles blancos aislados) fuera del objeto principal sin afectar significativamente su tamaño general.³⁶
- **Cierre (Closing):** Se aplica una operación de dilatación seguida de una de erosión. Este proceso es ideal para llenar pequeños agujeros o huecos dentro de la máscara del objeto, creando una región sólida y conectada.³⁶

La aplicación secuencial de estas operaciones garantiza que la máscara que pasa a la siguiente etapa sea una representación robusta y topológicamente correcta del salmón.

Etapa 2: Extracción de la Línea Central mediante Esqueletización

Con una máscara limpia, el siguiente desafío es encontrar la línea central a lo largo del cuerpo del pez. La técnica principal para lograr esto es la **Transformada del Eje Medial**, comúnmente conocida como **esqueletización**.³⁸

El algoritmo de esqueletización reduce una forma binaria a una representación de 1 píxel de ancho que se encuentra equidistante de los bordes más cercanos del contorno. Este "esqueleto" preserva la topología y la extensión del objeto original y representa la línea central que se necesita medir.³⁹ En Python, la implementación más robusta y estándar para esta tarea se encuentra en la biblioteca scikit-image, específicamente la función skimage.morphology.skeletonize.³⁹

Etapa 3: Poda del Esqueleto (Post-procesamiento)

Un efecto secundario común de los algoritmos de esqueletización es la generación de pequeñas ramas espurias que se extienden desde la línea central principal. Estas ramas suelen aparecer en puntos de alta curvatura o donde el contorno de la máscara presenta ligeras irregularidades, incluso después del refinamiento.³⁹ Si no se eliminan, estas ramas añadirán longitud extra al esqueleto, corrompiendo la medición final.

La estrategia para eliminar estos artefactos se conoce como **poda (pruning)**. Un enfoque

robusto es tratar el esqueleto como un grafo, donde cada píxel blanco es un nodo y las conexiones con los píxeles vecinos son aristas.⁴¹ El proceso de poda implica:

1. Identificar todos los puntos finales (endpoints) del esqueleto, que son píxeles con un solo vecino.
2. Para cada punto final, trazar la rama a la que pertenece hasta llegar a un punto de bifurcación (un píxel con más de dos vecinos).
3. Medir la longitud de esta rama.
4. Si la longitud de la rama es inferior a un umbral predefinido (por ejemplo, 5-10 píxeles), se elimina del esqueleto.

Este proceso asegura que solo la espina dorsal principal del salmón permanezca, proporcionando una línea limpia para la medición.⁴² La biblioteca skan de Python está diseñada específicamente para el análisis de esqueletos y proporciona herramientas potentes para este tipo de análisis basado en grafos.⁴³

Etapa 4: Cálculo de la Longitud

Una vez que se obtiene un esqueleto limpio y podado, el cálculo de la longitud es sencillo. Se recorre el esqueleto píxel a píxel, sumando las distancias euclidianas entre píxeles consecutivos. La distancia entre dos píxeles adyacentes es 1 si el paso es en dirección cardinal (horizontal o vertical) y $\sqrt{2}$ si el paso es en diagonal. La suma de estas pequeñas distancias a lo largo de toda la trayectoria del esqueleto proporciona una medida muy precisa de la longitud curva del salmón.

2.3 Análisis Comparativo de las Técnicas de Medición

La elección entre la estimación de pose y la segmentación de instancia tiene profundas implicaciones en la calidad y versatilidad de las mediciones. La siguiente tabla resume las diferencias clave, proporcionando una base clara para la toma de decisiones.

Característica	Estimación de Pose (Puntos Clave)	Segmentación de Instancia (Máscaras)	Justificación y Racionalización
Granularidad del Resultado	Dispersa (puntos discretos)	Densa (máscara a nivel de píxel)	La pose proporciona una abstracción esquelética ² , mientras que la segmentación ofrece la forma completa y rellena.[14, 15]
Versatilidad de la Medición	Limitada (principalmente	Alta (longitud curva, anchura, área, proxy	Los puntos clave son ideales para distancias

	(longitudes entre puntos)	de biomasa)	lineales predefinidas.[30] Las máscaras permiten un análisis morfológico complejo, incluyendo la anchura en cualquier punto a lo largo del esqueleto y el área total.[21, 44]
Precisión para Objetos Curvos	Moderada a Baja (Aproximación)	Alta (Medición Directa)	La estimación de pose aproxima las curvas con segmentos de línea.[5] La esqueletización de una máscara deriva el verdadero eje medial, proporcionando una trayectoria mucho más precisa para la medición de la longitud.[38, 45]
Complejidad de la Anotación	Alta (Requiere la colocación precisa y consistente de muchos puntos)	Moderada (Requiere trazar un contorno)	Anotar muchos puntos clave con precisión es tedioso y subjetivo.[46] Aunque trazar máscaras lleva tiempo, las herramientas modernas lo han hecho eficiente, y es menos ambiguo que colocar un punto exacto para el "centro del ojo".
Coste Computacional	Menor (predice un pequeño vector de coordenadas)	Mayor (predice una máscara completa $m \times m$ para cada objeto)	La rama de predicción de máscaras añade un coste computacional en comparación con la predicción de solo coordenadas de puntos clave.[14, 25]
Robustez ante la Oclusión	Baja (la oclusión de un punto clave crítico)	Moderada a Alta (aún puede derivar una	Los métodos basados en segmentación son

	puede causar un fallo total)	forma parcial de las partes visibles)	inherentemente más robustos a la oclusión parcial porque razonan sobre parches visibles de un objeto.[47, 48]
--	------------------------------	---------------------------------------	---

La decisión arquitectónica se reduce a un compromiso entre la simplicidad de la medición y la fidelidad de la representación. Mientras que la estimación de pose ofrece una vía más directa para obtener una aproximación de la longitud, la segmentación de instancia, a pesar de requerir un pipeline de post-procesamiento más complejo, proporciona una base de datos mucho más rica y precisa. Para un objetivo que exige una estimación precisa de las "dimensiones" de un animal no rígido, la fidelidad de la representación de la máscara es una ventaja fundamental que supera la complejidad añadida de su procesamiento.

Sección 3: Integración de la Visión Estéreo 3D para una Precisión Métrica Real

Hasta ahora, el análisis se ha centrado en la medición en el espacio de la imagen 2D, donde las unidades son píxeles. Sin embargo, para obtener mediciones del mundo real (en metros o centímetros), es necesario introducir la tercera dimensión. El uso de una cámara estéreo ZED, como la especificada en el proyecto, es la clave para pasar de mediciones relativas en píxeles a mediciones métricas absolutas.

3.1 La Ventaja de la Cámara ZED: De Píxeles a Nubes de Puntos

Los sistemas de medición 2D tradicionales se enfrentan a un problema fundamental: la ambigüedad de la escala. Un objeto parece más grande en píxeles cuanto más cerca está de la cámara. Para resolver esto, a menudo se requiere un objeto de referencia de tamaño conocido en la escena para calibrar una relación de "píxeles por métrica".⁴⁹ Este enfoque es inflexible, propenso a errores y poco práctico en entornos dinámicos como una granja de peces.

La visión estéreo elimina por completo esta limitación. Una cámara ZED utiliza sus dos sensores para calcular la disparidad y generar un **mapa de profundidad denso**. Este mapa es una imagen donde el valor de cada píxel (u, v) no es un color, sino una medida de la distancia Z desde ese punto de la escena hasta la cámara.⁵⁰ Esta información de profundidad por píxel es la clave para la reconstrucción 3D.

La Fórmula de Proyección

La conversión de una coordenada de píxel 2D (u, v) con su profundidad Z asociada a una coordenada del mundo real 3D (X, Y, Z) se realiza mediante las ecuaciones del modelo de cámara estenopeica (pinhole camera). Estas ecuaciones utilizan los **parámetros intrínsecos** de la cámara, que son la distancia focal (f_x, f_y) y el punto principal (c_x, c_y):

$$X = \frac{(u - c_x)}{f_x} \times Z$$

$$Y = \frac{(v - c_y)}{f_y} \times Z$$

$$Z = Z$$

Estos parámetros intrínsecos son únicos para cada cámara y se determinan durante el proceso de calibración.

Uso del SDK pyzed

Afortunadamente, no es necesario implementar estas ecuaciones desde cero. El SDK de ZED, a través de su envoltura de Python pyzed, proporciona funciones de alto nivel para realizar esta conversión de manera eficiente. La función `zed.retrieve_measure(point_cloud, sl.MEASURE.XYZ)` recupera una nube de puntos 3D completa de la escena, donde cada punto corresponde a un píxel de la imagen. Alternativamente, la función `point_cloud.get_value(u, v)` puede ser utilizada para consultar la coordenada 3D (X, Y, Z) para un píxel específico (u, v) de la imagen.⁵¹ Esto simplifica enormemente la transición del dominio 2D al 3D.

3.2 Flujo de Trabajo para la Medición en 3D

Con la capacidad de obtener coordenadas 3D para cualquier píxel, ahora podemos definir los flujos de trabajo de medición 3D tanto para la estimación de pose como para la segmentación de instancia.

Desde la Estimación de Pose

1. **Predicción 2D:** El modelo YOLOv8-Pose predice un conjunto de puntos clave 2D $[(u_1, v_1), (u_2, v_2), \dots, (u_n, v_n)]$ para un salmón.
2. **Proyección 3D:** Para cada punto clave (u_i, v_i) , se consulta al SDK de ZED para obtener su correspondiente coordenada 3D del mundo real (X_i, Y_i, Z_i) . Esto transforma el esqueleto 2D en un esqueleto 3D.
3. Cálculo de Longitud 3D: La longitud total curva en 3D se calcula como la suma de las distancias euclidianas 3D entre los puntos clave 3D consecutivos:

$$L_{3D} = \sum_{i=1}^{n-1} \sqrt{(X_{i+1} - X_i)^2 + (Y_{i+1} - Y_i)^2 + (Z_{i+1} - Z_i)^2}$$

Este método proporciona una medida métrica real, pero su precisión sigue estando limitada por la naturaleza aproximada del esqueleto basado en puntos clave.

Desde la Segmentación de Instancia

Este flujo de trabajo, aunque más complejo, ofrece una precisión y versatilidad inigualables.

1. **Predicción 2D:** El modelo YOLOv8-Seg predice una máscara de segmentación 2D para el salmón.
2. **Procesamiento Morfológico 2D:** La máscara se refina, esqueletiza y poda utilizando el pipeline descrito en la Sección 2.2. El resultado es una línea central 2D limpia, representada como una lista ordenada de píxeles $[(u_1, v_1), (u_2, v_2), \dots, (u_m, v_m)]$.
3. **Proyección 3D de la Línea Central:** Se inicializa una lista para almacenar los puntos 3D. Se itera a través de cada píxel (u_i, v_i) de la línea central 2D. Para cada píxel, se utiliza `point_cloud.get_value(u_i, v_i)` del SDK de ZED para obtener su coordenada 3D (X_i, Y_i, Z_i) . El resultado es una **nube de puntos de la línea central 3D** del salmón.
4. **Cálculo de Longitud 3D:** La longitud total 3D se calcula sumando las distancias euclidianas entre los puntos 3D consecutivos en esta trayectoria 3D. Este método mide la longitud real a lo largo del cuerpo curvado del pez en el espacio tridimensional, lo que lo convierte en el enfoque más preciso posible con esta tecnología.

La combinación de la segmentación de instancia con una cámara de profundidad permite la creación de un modelo 3D parcial del objeto de interés. Al proyectar no solo los píxeles del esqueleto, sino todos los píxeles dentro de la máscara de segmentación, se genera una **nube de puntos segmentada** que representa la superficie visible del salmón.⁵³ Esta nube de puntos es una forma de "gemelo digital" del pez en ese instante.

A partir de esta rica representación 3D, se pueden calcular dimensiones adicionales más allá de la longitud. Por ejemplo, se pueden analizar secciones transversales de la nube de puntos para estimar el perímetro o la anchura del pez en diferentes puntos a lo largo de su cuerpo. El volumen total de la nube de puntos puede servir como un proxy para la biomasa, una métrica de gran valor en la acuicultura.⁵⁶ Esta capacidad de realizar un análisis morfométrico 3D completo hace que el enfoque de segmentación sea mucho más extensible y valioso para futuras preguntas de investigación en comparación con el enfoque de estimación de pose, que solo proporciona un esqueleto 1D disperso. Esta extensibilidad es una consideración arquitectónica clave que favorece fuertemente la segmentación.

Sección 4: Una Hoja de Ruta Técnica Recomendada para su Proyecto

Tras un análisis exhaustivo de las metodologías, las técnicas de medición y la integración 3D, es posible sintetizar toda la información en una recomendación clara y una hoja de ruta de implementación concreta. Este plan está diseñado para guiar el desarrollo del proyecto desde la etapa actual hasta un sistema funcional y validado.

4.1 Evaluación Directa y Recomendación Final

El análisis previo permite responder directamente a las preguntas centrales planteadas en la consulta inicial.

Respondiendo a las Preguntas Clave

1. "¿Es la estimación de pose la técnica correcta?"

No, no es la técnica óptima para este objetivo específico. Aunque la estimación de pose puede aproximar la longitud, es fundamentalmente menos precisa para cuerpos no rígidos y curvados. Su dependencia de un conjunto disperso de puntos la hace vulnerable a occlusiones y menos versátil para medir otras dimensiones biométricas como la anchura o el área.²⁹

2. "¿Dónde entra en juego un mecanismo de segmentación?"

La segmentación de instancia es la tecnología central y habilitadora para este proyecto. No debe considerarse un mecanismo auxiliar, sino el pilar fundamental del pipeline de medición. Proporciona la representación de datos más rica y fiel de la forma del salmón, lo que permite la medición más precisa y robusta de su longitud curva y abre la puerta a análisis morfométricos más avanzados.

Recomendación Arquitectónica

La arquitectura recomendada es un pipeline centrado en la **segmentación de instancia con YOLOv8-Seg**, seguido por el flujo de trabajo morfológico de **Refinar-Esqueletizar-Podar**, y finalmente, la **proyección y medición 3D** utilizando los datos de profundidad de la cámara ZED. Este enfoque ofrece la mayor precisión, robustez y extensibilidad futura.

La integración de este pipeline con el sistema de seguimiento DeepSORT, que ya se está utilizando, crea una sinergia poderosa. El modelo YOLOv8-Seg proporciona detecciones de alta calidad (máscaras y cuadros delimitadores derivados) para cada fotograma. DeepSORT utiliza esta información para asignar y mantener una identidad temporal (ID de seguimiento) a cada salmón a lo largo del tiempo.⁵⁷ Al combinar ambos, el sistema puede rastrear a un individuo específico y registrar sus dimensiones a medida que se mueve y flexiona su cuerpo. Esto eleva el proyecto de una simple "herramienta de medición" a una "plataforma de análisis biométrico y de comportamiento", capaz de realizar un seguimiento longitudinal del crecimiento o estudiar la dinámica de la natación.

4.2 Plan de Implementación Paso a Paso

A continuación se detalla un plan de acción para desarrollar e implementar la arquitectura

recomendada.

1. Curación y Anotación del Conjunto de Datos

- **Fuentes de Datos:** El rendimiento del modelo dependerá en gran medida de la calidad y representatividad del conjunto de datos de entrenamiento. Se pueden explorar fuentes públicas como Kaggle⁵⁸ y Roboflow Universe⁶⁰ para obtener imágenes iniciales. Sin embargo, es muy probable que se necesite crear un conjunto de datos personalizado utilizando los archivos de video .svo capturados con la cámara ZED para que el modelo se generalice bien al entorno de despliegue específico.
- **Herramientas de Anotación:** Se deben anotar los fotogramas de video para la tarea de **segmentación de instancia**. Herramientas como CVAT (Computer Vision Annotation Tool)⁶¹ o la plataforma de anotación de Roboflow son ideales para esta tarea. El objetivo es dibujar un polígono preciso alrededor de cada salmón en una variedad de imágenes que capturen diferentes poses, condiciones de iluminación y densidades de población. El resultado de este proceso será un conjunto de datos con máscaras para cada salmón individual.

2. Entrenamiento y Validación del Modelo

- **Entrenamiento:** Utilizar el conjunto de datos anotado para entrenar un modelo **YOLOv8-Seg**. Se puede comenzar con un modelo pre-entrenado en el conjunto de datos COCO (por ejemplo, yolov8n-seg.pt) y ajustarlo finamente (fine-tuning) con los datos de salmones. El marco de trabajo de Ultralytics simplifica este proceso, requiriendo un archivo YAML que defina las rutas del conjunto de datos y los parámetros de entrenamiento.¹⁰
- **Validación:** Durante el entrenamiento, el rendimiento del modelo debe ser monitoreado en un conjunto de validación separado para evitar el sobreajuste. Las métricas clave a observar son la Precisión Media Promedio (mAP) para la detección de cuadros y máscaras.

3. Integración del Pipeline (Nivel de Código)

- **Modificación del Script Existente:** El script de Python actual que procesa los fotogramas de la cámara ZED debe ser modificado. El bucle principal que captura un nuevo fotograma debe ahora pasar la imagen al modelo YOLOv8-Seg entrenado para la inferencia.
- **Procesamiento de Resultados:** El resultado de la inferencia del modelo contendrá cuadros delimitadores, identificadores de clase, puntuaciones de confianza y, lo más

importante, las **máscaras de segmentación**.⁶²

- **Integración con DeepSORT:** Los cuadros delimitadores (que se pueden derivar fácilmente de las máscaras) se pasarán al rastreador DeepSORT. DeepSORT asociará estas detecciones a través de los fotogramas y asignará un ID de seguimiento único y persistente a cada salmón.⁵⁷

4. Desarrollo del Módulo de Medición 3D (Python)

Este módulo será una nueva función o clase en el código que toma el ID de seguimiento, la máscara 2D y la nube de puntos 3D del fotograma actual como entrada.

- Para cada salmón rastreado con un ID único y su máscara correspondiente:
 1. **Refinar Máscara:** Aplicar cv2.morphologyEx con las operaciones MORPH_OPEN y MORPH_CLOSE para limpiar la máscara.³⁶
 2. **Esqueletizar:** Utilizar skimage.morphology.skeletonize para obtener la línea central 2D.⁴⁰
 3. **Podar Esqueleto:** Implementar un algoritmo de poda para eliminar las ramas espurias. Un enfoque basado en grafos utilizando la biblioteca networkx para analizar la conectividad de los píxeles del esqueleto es una solución robusta.⁴¹
 4. **Proyectar a 3D:** Inicializar una lista vacía para los puntos 3D. Iterar a través de cada coordenada de píxel (u, v) del esqueleto 2D podado. Para cada píxel, usar la función point_cloud.get_value(u, v) del SDK pyzed para obtener la coordenada 3D (X, Y, Z) y añadirla a la lista.⁵¹
 5. **Calcular Longitud:** Calcular la longitud total sumando las distancias euclidianas entre los puntos 3D consecutivos en la lista.
 6. **Almacenar Resultados:** Almacenar la longitud calculada junto con el ID de seguimiento y la marca de tiempo del fotograma.

5. Calibración y Validación del Sistema

- **Calibración Intrínseca:** El SDK de ZED maneja la calibración intrínseca de la cámara, proporcionando los parámetros necesarios para la proyección 3D.
- **Validación de Extremo a Extremo:** Es crucial validar la precisión de todo el sistema de medición.
 - **Procedimiento:** Colocar un objeto flexible de longitud conocida (por ejemplo, una cuerda marcada o una regla flexible) en el agua. Grabar este objeto de calibración a varias distancias y orientaciones con la cámara ZED.
 - **Evaluación:** Ejecutar el pipeline completo sobre estos videos de calibración. Comparar la longitud 3D calculada por el sistema con la longitud real conocida del objeto.
 - **Análisis de Errores:** Calcular el error absoluto y relativo promedio. Este proceso

cuantificará la precisión del sistema en el mundo real y ayudará a identificar posibles fuentes de error sistemático.

Siguiendo esta hoja de ruta, el proyecto puede pasar de su fase conceptual a un sistema de medición biométrica 3D robusto, preciso y validado, basado en una arquitectura de visión por computadora sólida y bien justificada.

Obras citadas

1. pmc.ncbi.nlm.nih.gov, fecha de acceso: octubre 30, 2025,
<https://pmc.ncbi.nlm.nih.gov/articles/PMC11200986/#:~:text=Animal%20pose%20estimation%20involves%20identifying,pose%20estimation%20methods%20for%20animals>.
2. Towards Multi-Modal Animal Pose Estimation: An In-Depth Analysis - arXiv, fecha de acceso: octubre 30, 2025, <https://arxiv.org/html/2410.09312v1>
3. Real-Time Pose Estimation in Computer Vision - Viso Suite, fecha de acceso: octubre 30, 2025,
<https://viso.ai/deep-learning/pose-estimation-ultimate-overview/>
4. Animal Pose Estimation Based on 3D Priors - MDPI, fecha de acceso: octubre 30, 2025, <https://www.mdpi.com/2076-3417/13/3/1466>
5. Benchmarking Fish Dataset and Evaluation Metric in Keypoint Detection - Towards Precise Fish Morphological Assessment in Aquaculture Breeding - IJCAI, fecha de acceso: octubre 30, 2025,
<https://www.ijcai.org/proceedings/2024/0816.pdf>
6. arXiv:2409.08695v3 [cs.CV] 25 Sep 2024, fecha de acceso: octubre 30, 2025,
<https://www.arxiv.org/pdf/2409.08695>
7. Improved YOLOv8-Pose Algorithm for Albacore Tuna (*Thunnus alalunga*) Fork Length Extraction and Weight Estimation - ResearchGate, fecha de acceso: octubre 30, 2025,
https://www.researchgate.net/publication/380454476_Improved_YOLOv8-Pose_Algorithm_for_Albacore_Tuna_Thunnus_alalunga_Fork_Length_Extraction_and_Weight_Estimation
8. Multi-animal pose estimation, identification and tracking with DeepLabCut - PMC - NIH, fecha de acceso: octubre 30, 2025,
<https://pmc.ncbi.nlm.nih.gov/articles/PMC9007739/>
9. Implementation of YOLOv8-Pose Model for Identification and Estimation of the Length of Skipjack and Tuna Like Species on the Web, fecha de acceso: octubre 30, 2025,
https://www.bio-conferences.org/articles/bioconf/pdf/2024/25/bioconf_icms2024_01005.pdf
10. Pose Estimation Datasets Overview - Ultralytics YOLOv8 Docs #2517 - GitHub, fecha de acceso: octubre 30, 2025,
<https://github.com/orgs/ultralytics/discussions/2517>
11. (PDF) Implementation of YOLOv8-Pose Model for Identification and Estimation of the Length of Skipjack and Tuna Like Species on the Website - ResearchGate, fecha de acceso: octubre 30, 2025,

https://www.researchgate.net/publication/380334762_Implementation_of_YOLOv8-Pose_Model_for_Identification_and_Estimation_of_the_Length_of_Skipjack_and_Tuna_Like_Species_on_the_Website

12. Implementation of YOLOv8-Pose Model for Identification and Estimation of the Length of Skipjack and Tuna Like Species on the Website - Semantic Scholar, fecha de acceso: octubre 30, 2025,
<https://www.semanticscholar.org/paper/Implementation-of-YOLOv8-Pose-Model-for-and-of-the-Tanbir-Iqbal/8080ea9bbad555a9019bd9f29871f2f225f44e97>
13. DeepPoseKit, a software toolkit for fast and robust animal pose estimation using deep learning | eLife, fecha de acceso: octubre 30, 2025,
<https://elifesciences.org/articles/47994>
14. Instance Segmentation in Practice: Techniques, Tools, and Applications - Lightly AI, fecha de acceso: octubre 30, 2025,
<https://www.lightly.ai/blog/instance-segmentation>
15. What Is Instance Segmentation? [2024 Guide & Tutorial] - V7 Go, fecha de acceso: octubre 30, 2025,
<https://www.v7labs.com/blog/instance-segmentation-guide>
16. How is object segmentation and object detection are different? - ResearchGate, fecha de acceso: octubre 30, 2025,
https://www.researchgate.net/post/How_is_object_segmentation_and_object_detection_are_different
17. pmc.ncbi.nlm.nih.gov, fecha de acceso: octubre 30, 2025,
<https://pmc.ncbi.nlm.nih.gov/articles/PMC12159697/#:~:text=Instance%20segmentation%20combines%20the%20principles.of%20objects%20within%20predefined%20classes.>
18. What Is Instance Segmentation? - IBM, fecha de acceso: octubre 30, 2025,
<https://www.ibm.com/think/topics/instance-segmentation>
19. Instance Segmentation: A Quick Guide - Ultralytics, fecha de acceso: octubre 30, 2025,
<https://www.ultralytics.com/blog/what-is-instance-segmentation-a-quick-guide>
20. Image-based, unsupervised estimation of fish size from commercial landings using deep learning - SciSpace, fecha de acceso: octubre 30, 2025,
<https://scispace.com/pdf/image-based-unsupervised-estimation-of-fish-size-from-37si2mbmn9.pdf>
21. (PDF) Automatic measurement of fish from images using convolutional neural networks, fecha de acceso: octubre 30, 2025,
https://www.researchgate.net/publication/379924418_Automatic_measurement_of_fish_from_images_using_convolutional_neural_networks
22. Image Segmentation with Mask R-CNN, GrabCut, and OpenCV - GeeksforGeeks, fecha de acceso: octubre 30, 2025,
<https://www.geeksforgeeks.org/computer-vision/image-segmentation-with-mask-r-cnn-grabcut-and-opencv/>
23. (PDF) An enhanced YOLOv8 algorithm for fish school recognition integrating SEAttention and LDConv - ResearchGate, fecha de acceso: octubre 30, 2025,
https://www.researchgate.net/publication/396498836_An_enhanced_YOLOv8_alg

orithm_for_fish_school_recognition_integrating_SEAttention_and_LDConv

24. Comparative Study of Fish Detection and Classification Performance Using the YOLOv8-Seg Model - ResearchGate, fecha de acceso: octubre 30, 2025,
[https://www.researchgate.net/publication/383108714 Comparative Study of Fish Detection and Classification Performance Using the YOLOv8-Seg Model](https://www.researchgate.net/publication/383108714_Comparative_Study_of_Fish_Detection_and_Classification_Performance_Using_the_YOLOv8-Seg_Model)
25. Review: Mask R-CNN (Instance Segmentation & Human Pose Estimation) | by Sik-Ho Tsang | Analytics Vidhya | Medium, fecha de acceso: octubre 30, 2025,
<https://medium.com/analytics-vidhya/review-mask-r-cnn-instance-segmentation-human-pose-estimation-61080a93bf4>
26. Automatic deep learning-based pipeline for Mediterranean fish segmentation - Frontiers, fecha de acceso: octubre 30, 2025,
<https://www.frontiersin.org/journals/marine-science/articles/10.3389/fmars.2025.1525524/full>
27. Application of Improved YOLOv8n-seg in Crayfish Trunk Segmentation - Israeli Journal of Aquaculture - Bamidgeh, fecha de acceso: octubre 30, 2025,
<https://ija.scholasticahq.com/api/v1/articles/126850-application-of-improved-yolo-v8n-seg-in-crayfish-trunk-segmentation.pdf>
28. Learning to Measure Invisible Fish - DiVA portal, fecha de acceso: octubre 30, 2025, <https://www.diva-portal.org/smash/get/diva2:1677704/FULLTEXT01.pdf>
29. A Dual-Segmentation Framework for the Automatic Detection and Size Estimation of Shrimp, fecha de acceso: octubre 30, 2025,
<https://www.mdpi.com/1424-8220/25/18/5830>
30. (PDF) Key point detection method for fish size measurement based on deep learning, fecha de acceso: octubre 30, 2025,
[https://www.researchgate.net/publication/373940139 Key_point_detection_method_for_fish_size_measurement_based_on_deep_learning](https://www.researchgate.net/publication/373940139_Key_point_detection_method_for_fish_size_measurement_based_on_deep_learning)
31. Fish Keypoints Detection for Ecology Monitoring Based on Underwater Visual Intelligence, fecha de acceso: octubre 30, 2025,
<https://www.semanticscholar.org/paper/Fish-Keypoints-Detection-for-Ecology-Monitoring-on-Suo-Huang/557c960ae15779dd16cbb0e0f09a0fe9894526eb>
32. Fish keypoint detection for offshore aquaculture: a robust deep learning approach with PCA-based shape constraint - Frontiers, fecha de acceso: octubre 30, 2025,
<https://www.frontiersin.org/journals/marine-science/articles/10.3389/fmars.2025.619457/full>
33. Image segmentation using Morphological operations in Python - GeeksforGeeks, fecha de acceso: octubre 30, 2025,
<https://www.geeksforgeeks.org/python/image-segmentation-using-morphological-operation/>
34. Python OpenCV - Morphological Operations - GeeksforGeeks, fecha de acceso: octubre 30, 2025,
<https://www.geeksforgeeks.org/python/python-opencv-morphological-operations/>
35. OpenCV Morphological Operations - PylImageSearch, fecha de acceso: octubre 30, 2025,

- <https://pyimagesearch.com/2021/04/28/opencv-morphological-operations/>
36. Morphological Transformations - OpenCV, fecha de acceso: octubre 30, 2025, https://docs.opencv.org/4.x/d9/d61/tutorial_py_morphological_ops.html
37. Biomedical Image Analysis Part 3— Mask Creation, Dilation, Erosion, and Image Filtering | by Serra Aksoy (SeruRays) | Medium, fecha de acceso: octubre 30, 2025, <https://medium.com/@serurays/biomedical-image-analysis-part-3-mask-creation-dilation-erosion-and-image-filtering-b667e3ca568e>
38. Morphology – Skeletonization/Medial Axis Transform, fecha de acceso: octubre 30, 2025, <https://homepages.inf.ed.ac.uk/rbf/HIPR2/skeleton.htm>
39. Skeletonize — skimage 0.25.2 documentation, fecha de acceso: octubre 30, 2025, https://scikit-image.org/docs/0.25.x/auto_examples/edges/plot_skeleton.html
40. skimage.morphology — skimage 0.25.2 documentation, fecha de acceso: octubre 30, 2025, <https://scikit-image.org/docs/0.25.x/api/skimage.morphology.html>
41. Removing branches from image skeleton : r/computervision - Reddit, fecha de acceso: octubre 30, 2025, https://www.reddit.com/r/computervision/comments/10lspjl/removing_branches_from_image_skeleton/
42. (A) Removal of additional branches with the SkelTre skeleton graph. (B)... - ResearchGate, fecha de acceso: octubre 30, 2025, https://www.researchgate.net/figure/A-Removal-of-additional-branches-with-the-SkelTre-skeleton-graph-B-Vertices-with_fig9_265474151
43. Feature request: Add function to prune skeleton · Issue #97 · jni/skan - GitHub, fecha de acceso: octubre 30, 2025, <https://github.com/jni/skan/issues/97>
44. Measuring size of objects in an image with OpenCV - PyImageSearch, fecha de acceso: octubre 30, 2025, <https://pyimagesearch.com/2016/03/28/measuring-size-of-objects-in-an-image-with-opencv/>
45. How to Use OpenCV with ZED in Python - Stereolabs, fecha de acceso: octubre 30, 2025, <https://www.stereolabs.com/docs/opencv/python>
46. Tutorial – Using Depth Perception – Stereolabs, fecha de acceso: octubre 30, 2025, <https://www.stereolabs.com/docs/tutorials/depth-sensing>
47. How to convert zed camera pixel coordinates (u, v) to world coordinates (X, Y, Z) ?, fecha de acceso: octubre 30, 2025, <https://community.stereolabs.com/t/how-to-convert-zed-camera-pixel-coordinates-u-v-to-world-coordinates-x-y-z/3929>
48. (PDF) A Measurement Model for Aquatic Animals Based on Instance Segmentation and 3D Point Cloud - ResearchGate, fecha de acceso: octubre 30, 2025, https://www.researchgate.net/publication/384552953_A_Measurement_Model_for_Aquatic_Animals_Based_on_Instance_Segmentation_and_3D_Point_Cloud
49. Aligning a point cloud of 2D Mask that was generated from depth camera in ZED, fecha de acceso: octubre 30, 2025, <https://stackoverflow.com/questions/79415833/aligning-a-point-cloud-of-2d-mas>

[k-that-was-generated-from-depth-camera-in-zed](#)

50. 3D point cloud object segmentation based on sensor fusion and 2D mask guidance, fecha de acceso: octubre 30, 2025,
<https://developer.supervisely.com/getting-started/python-sdk-tutorials/point-clouds/point-cloud-segmentation-with-2d-mask-guidance>
51. A Non-Contact Measurement of Animal Body Size Based on Structured Light, fecha de acceso: octubre 30, 2025,
https://www.researchgate.net/publication/377612974_A_Non-Contact_Measurement_of_Animal_Body_Size_Based_on_Structured_Light
52. MuhammadMoinFaisal/YOLOv8-DeepSORT-Object-Tracking - GitHub, fecha de acceso: octubre 30, 2025,
<https://github.com/MuhammadMoinFaisal/YOLOv8-DeepSORT-Object-Tracking>
53. Fish Dataset - Kaggle, fecha de acceso: octubre 30, 2025,
<https://www.kaggle.com/datasets/mahmoodyousaf/fish-dataset>
54. Fish Dataset Image Segmentation using U-Net - Kaggle, fecha de acceso: octubre 30, 2025,
<https://www.kaggle.com/code/mayur7garg/fish-dataset-image-segmentation-using-u-net>
55. Kaggle-Fish-Detection Instance Segmentation Dataset by Innoweave - Roboflow Universe, fecha de acceso: octubre 30, 2025,
<https://universe.roboflow.com/innoweave/kaggle-fish-detection-o8ghb>
56. Object Detection and Tracking using Yolov8 and DeepSORT | by Serra Aksoy (SeruRays), fecha de acceso: octubre 30, 2025,
<https://medium.com/@serurays/object-detection-and-tracking-using-yolov8-and-deepsort-47046fc914e9>
57. models/yolov8/ · ultralytics · Discussion #10285 - GitHub, fecha de acceso: octubre 30, 2025, <https://github.com/orgs/ultralytics/discussions/10285>
58. Instance segmentation with yolov8 · Issue #1330 - GitHub, fecha de acceso: octubre 30, 2025, <https://github.com/ultralytics/ultralytics/issues/1330>
59. Track - Ultralytics YOLOv8 Docs #2687 - GitHub, fecha de acceso: octubre 30, 2025, <https://github.com/orgs/ultralytics/discussions/2687>