

# PROPUESTA DE SISTEMA DE CONTEO DE ALEVINES DE TILAPIA ROJA DE BAJO COSTO USANDO TÉCNICAS DE VISIÓN ARTIFICIAL

Rubén Darío Vásquez Salazar<sup>1</sup>, Ahmed Alejandro Cardona Mesa<sup>2</sup>, Leidy Yohana Ocampo Osorio<sup>3</sup>

<sup>1</sup> M.Sc. en Ingeniería – Automatización Industrial. Docente. E-mail: rdvasquez@elpoli.edu.co

<sup>2</sup> Estudiante Ingeniería en Instrumentación y Control. E-mail: ahmed\_cardona04072@elpoli.edu.co

<sup>3</sup> Estudiante Ingeniería en Instrumentación y Control. E-mail: leidy\_ocampo91091@elpoli.edu.co

<sup>1,2,3</sup> Grupo de Investigación ICARO. Facultad de Ingenierías. Politécnico Colombiano Jaime Isaza Cadavid. Medellín - Colombia

## RESUMEN

En el país la comercialización de alevines se realiza por unidades y no por peso y, al no existir un equipo automático, requiere que su conteo se realice de forma manual. La actual propuesta consiste en una rampa con inclinación variable, que permita regular la velocidad de descenso. Un primer prototipo no cuenta con iluminación, lo cual generó problemas por las diferentes condiciones de luz de operación, lo que requirió de un diseño de un sistema cerrado con iluminación interna propia. El software está diseñado bajo LabVIEW, con capacidad de 400 alevines/segundo, convirtiendo la cámara en un limitante por su velocidad de 30fps. En el presente artículo se explican los diferentes algoritmos de visión utilizados, en línea con el video de la cámara. Al final se presentan resultados, lográndose el conteo con un error inferior al 10%, y proponiendo mejoras en el sistema como el uso de tecnología que no dependa de computador y equipos de mejores especificaciones, que aumentarían el costo pero mejorarían la precisión.

**Palabras clave:** Alevines, tilapia roja, visión artificial, procesamiento de imágenes, LabVIEW, conteo automático.

Recibido: 08 de Septiembre de 2015

Aceptado: 15 de Diciembre de 2015

Received: September 8th, 2015

Accepted: December 15th, 2015

## A PROPOSAL OF LOW COST SYSTEM FOR COUNTING RED TILAPIA FINGERLINGS BY USING ARTIFICIAL VISION TECHNIQUES

### ABSTRACT

In Colombia, the fish fingerling marketing is done by units and not by weight and, due to the lack of automatic equipment, it requires a manual counting. The current proposal is a ramp with variable inclination, in order to regulate the lowering speed. A first prototype had no lighting, which caused problems due to the different lighting operating conditions, which required the design of a closure with its own internal lighting. The software is designed under LabVIEW, with capacity of 400 fingerlings per second, becoming the video camera a limitation because of its rate of 30fps. In this article different vision algorithms used on line with the video camera are explained. The final results are presented, achieving a counting error of less than 6%, and some improvements in the system such as technology not dependent on computers and equipment with better specifications are suggested, which would increase the cost but at the same time would improve accuracy.

**Keywords:** Fingerlings, red tilapia, artificial vision, image processing, LabVIEW, automatic counting.

*Cómo citar este artículo:* R. D. Vásquez, A. A. Cardona y L. Y. Ocampo, "Propuesta de sistema de conteo de alevines de tilapia roja de bajo costo usando técnicas de visión artificial," *Revista Politécnica*, vol. 11, no. 21, pp. 85-95, 2015.

## 1. INTRODUCCIÓN

Colombia, debido a su ubicación geográfica y a los pisos térmicos de los cuales goza, posee unas condiciones naturales que permiten el desarrollo y sostenibilidad de diversas actividades agrícolas entre ellas la Acuicultura, la cual ha sido una de las actividades económicas con mayor desarrollo y crecimiento dentro del sector agropecuario [1]. La piscicultura en Colombia está representada por la producción de tilapia (*Oreochromis*), cachama (*Colossoma macropomum*), trucha (*Oncorhynchus mykiss*) y algunas especies nativas. Para el año 2011, la producción de tilapia representó el 65,21% de la piscicultura, la cachama 21,44%, la trucha 7,58% y las especies nativas con 5,62% [2].

Una de las principales actividades dentro de la piscicultura es la producción de alevines, como se designa a las crías recién nacidas de peces durante un periodo comprendido entre el nacimiento y el momento en que estos alcancen una determinada talla, dependiendo de la especie [3]. En esta etapa de alevinaje, es de gran importancia para el piscicultor realizar el conteo de los alevines, debido a que su comercialización se realiza por cantidad de animales. Además, para pasar a la siguiente etapa de levante y engorde dentro del proceso productivo, los peces se siembran en estanques, en donde se requiere conocer con la mayor precisión posible el número correcto de alevines cultivados, para determinar parámetros fundamentales como número de peces por metro cúbico de agua, cantidad de alimento suministrado, nivel de oxígeno y amonio en el agua, pH, entre otros. Todas estas variables deben ser medidas y controladas por el piscicultor para garantizar una buena calidad en el producto final.

Tradicionalmente existen diferentes técnicas de conteo utilizadas por los piscicultores y su selección depende principalmente de la cantidad de alevines producidos por la estación piscícola, los pequeños productores alcanzan volúmenes promedio por semana entre 2.000 y 3.000 alevines, hasta las grandes granjas piscícolas que alcanzan valores de 60.000 o más alevines de producción semanal. Las técnicas de conteo de alevines más conocidas son las siguientes:

a) Conteo de forma manual, el cual consiste en contar los peces manualmente uno por uno, pasándolos de un recipiente a otro, este método se realiza de forma lenta y se encuentra sujeto a los errores propios de la condición humana al

desarrollar una actividad monótona y repetitiva. b) Por volumen en seco, en el cual, con ayuda de una cuchara medidora de un volumen conocido, se logra sacar una relación volumen-número de los alevines, este método carece de precisión al verse afectado por los diferentes tamaños de alevines que se pueden encontrar en una población determinada. c) Por volumen en agua, en el cual se relaciona el volumen de agua desplazada con el número de alevines en un recipiente graduado, presentando una desventaja similar al método anterior. d) Por comparación visual, el cual consiste en determinar cantidades numéricas de alevines por medio de una comparación con una población conocida en un recipiente similar, este método presenta un alto error al determinarse la cantidad de peces de forma subjetiva. e) Gravimétrico, en el cual con la ayuda de una balanza electrónica y recipientes secos se logra encontrar una relación entre el peso en seco y el número de animales, originando un error al ser necesario considerar que todos los alevines de la población presentan exactamente el mismo tamaño y peso [4]. f) Conteo con un colador, para el cual ya se conoce aproximadamente la cantidad de peces que puede contener [5], al igual que en varios de los métodos presentados anteriormente, se ve afectado por las suposiciones inherentes de la comparación realizada frente a un patrón poco preciso. Los métodos expuestos a excepción del conteo de forma manual comparten el mismo inconveniente que les resta precisión al momento de determinar la cantidad correcta de alevines, debido a que no se puede garantizar un tamaño y peso homogéneo de estos, por lo que determinar la cantidad de peces se convierte en un proceso de cálculo aproximado y no en un conteo preciso. El conteo manual se realiza de forma directa y puede ser más confiable, aunque genera errores y dificultades por parte de la persona que realiza el proceso, como lo son lentitud, agotamiento y desconcentración al realizar conteos de cantidades elevadas de alevines por tiempo prolongado.

Los métodos tradicionales de conteo de alevines presentan una desventaja adicional, que consiste en la manipulación directa e invasiva de los peces, lo que ocasiona heridas superficiales sobre los mismos, contribuyendo en la aparición de hongos y enfermedades que disminuyen las características de calidad del producto final y aumentan la tasa de mortandad de alevines en esta etapa del proceso de producción piscícola. Un método menos invasivo dado que no requiere que el pez sea retirado del

agua se propone en [6], donde se utiliza tecnología infrarroja y comunicación inalámbrica para la transmisión de los datos; un problema con esta propuesta es que si dos o más alevines se desplazan por el rango de acción del sensor de forma simultánea, el conteo no se realizaría de forma adecuada; los alevines deben organizarse para formar una línea forzados por el diámetro del conducto y el último inconveniente es que el sensor es del tipo ON/OFF, impidiendo extraer más información como tamaño, color, fotografía, entre otros.

Durante el desarrollo del presente trabajo de investigación se describe un método para el conteo de alevines basado en un algoritmo de visión artificial, mediante el cual se realiza la adquisición de imágenes con ayuda de una cámara web de alta definición, se aplican técnicas de preprocesamiento y procesamiento de imágenes para obtener la información necesaria y posteriormente determinar la cantidad de alevines por medio de una interfaz gráfica de usuario desarrollada con el software LabVIEW. Se cuenta con un prototipo basado en una plataforma que permite el desplazamiento de los peces y la captura de las imágenes durante su recorrido.

## 2. MATERIALES Y MÉTODOS

### 2.1 LabVIEW

Es una plataforma de programación gráfica que ofrece herramientas para resolver diferentes problemas a nivel industrial, con capacidad para la innovación. Contiene aplicaciones para adquisición y análisis de datos, control de instrumentos, sistemas de monitoreo y embebidos, sistemas de pruebas automatizadas y validación [7]. Comparada con otras herramientas utilizadas en el control e instrumentación de procesos, tiene una inmensa fortaleza en la parte gráfica, debido a que tiene un panel frontal y una librería de símbolos de alto nivel, por lo que sobresale para la creación de interfaces hombre máquina de alta calidad. Permite programación de estructuras paralelas y optimiza el uso de procesadores presentes en la CPU del computador. La programación visual es otra característica importante, proporcionando un ambiente de programación más claro y fácil al realizar revisión o correcciones al código. En su diagrama de bloques, soporta múltiples tipos de datos, estructuras comunes de programación y librerías para comunicación. Su capacidad de

generar archivos ejecutables (run-time), permite que las aplicaciones diseñadas en LabVIEW se puedan ejecutar en otros computadores sin necesidad de adquirir licencias, siempre y cuando este ejecutable haya sido generado en un LabVIEW licenciado para tal tipo.

### 2.2 Visión Artificial

La visión artificial es una disciplina que engloba todos los procesos y elementos que proporcionan percepción visual a una máquina y se aplica en procesos industriales en los que la simple detección de presencia no resulta una fuente de información suficiente. Las estructuras y propiedades del mundo tridimensional que se quieren deducir en visión artificial incluyen no sólo sus propiedades geométricas, sino también sus propiedades materiales [8]. La visión es la ventana al mundo de muchos organismos. Su función principal es reconocer y localizar objetos en el ambiente mediante el procesamiento de las imágenes. La visión computacional es el estudio de estos procesos, para entenderlos y construir máquinas con capacidades similares [9]. La definición que se acerca más a la idea actual sobre visión computacional es la de [10] *“Visión es un proceso que produce a partir de las imágenes del mundo exterior una descripción que es útil para el observador y que no tiene información irrelevante”*.

Algunos ejemplos de algoritmos de visión computacional y aplicaciones son:

- Reconstrucción de estructuras en 3D a partir de cientos de fotografías que se superponen [11]
- Algoritmos que pueden construir un modelo detallado en 3D de fotografías con diferente exposición tomadas de internet [12]
- Rastreo de personas, de modo que se puede rastrear incluso si se tienen fondos congestionados [13]
- Detección de rostros en imágenes [14]
- Industria: Control de calidad.
- Robótica: Localización de objetos [15]

Para las necesidades de la industria actual se han desarrollado múltiples arquitecturas para la detección de defectos visuales en superficies planas y homogéneas, tales como textiles, maderas, aceros, aluminio, plásticos, pasta de papel y un sinnúmero de productos industriales, mostrando un gran interés tanto para la industria como para la investigación aplicada. [16]

### 2.3 Adquisición de imágenes

La primera etapa, dentro de un proceso de visión artificial es la etapa de captura y adquisición de la imagen, para ello se necesitan sensores generalmente ópticos con los que se obtiene información de una escena. Una correcta adquisición de la imagen supone un paso muy importante para que el proceso de reconocimiento tenga éxito. Dentro de esta etapa existen múltiples factores que afectan directamente al proceso de captura de la imagen, formados fundamentalmente por el hardware de visión artificial (cámara, tarjeta de adquisición de imagen, ordenador y software) y el entorno y posicionamiento de los elementos como la iluminación, el fondo, posición correcta de la cámara, ruido eléctrico-óptico externo, entre otros [8].

### 2.4 Máscara

Una vez adquirida la imagen por la cámara es necesario realizar el pre-procesamiento para eliminar ciertas regiones de la imagen que no contienen información de interés y facilitar el posterior procesamiento. Una máscara de imagen aísla partes de una imagen y determina si los píxeles de la imagen de inspección correspondientes a los píxeles de la máscara se procesan o no, si un píxel en la máscara de imagen tiene un valor diferente de cero en la imagen de inspección se procesa, de lo contrario no se procesa. El resultado será entonces una fracción de la imagen original.

### 2.5 Segmentación por RGB

La segmentación es el proceso mediante el cual la imagen se descompone en elementos que hacen parte de objetos, evaluando si cada píxel de la imagen pertenece o no al objeto de interés. La umbralización de imágenes de objetos en entornos automatizados muchas veces es compleja por la falta de conocimiento apriori del número de objetos a detectar, por la influencia de elementos no deseados como sombras, brillos, la complejidad de colores, texturas y tamaño de los objetos y la posibilidad de solapamiento de estos. Esto hace que la elección de un número bajo de umbral influya en la subsegmentación, se detectan menos regiones de las deseadas, o la elección de un número elevado lo haga en una sobresegmentación, se detectan más regiones de las que interesa. [17]

La segmentación de la imagen RGB se realiza determinando individualmente las componentes de

color rojo, verde y azul que conforman la imagen, seleccionando de un grupo de imágenes el umbral máximo y mínimo para cada uno de los componentes de color que pertenece al objeto de interés, garantizando que la imagen contenga la cantidad suficiente de información para su posterior análisis. La imagen resultante es de tipo binaria, en donde se asigna el valor de "1" a todos los píxeles que se encuentren dentro del rango de umbral seleccionado y el valor "0" para los que se encuentren por fuera de él, la selección del umbral dentro del proceso de segmentación ocasiona la aparición de ruidos en la imagen, que deben ser eliminados por otros métodos en las etapas siguientes.

### 2.6 Convex hull

Al obtener la imagen binarizada después del proceso de segmentación es recomendable definir la forma de los objetos de interés dentro de la imagen de inspección, por esto se usa Convex Hull que calcula los vértices de un polígono convexo, que son el conjunto de puntos que forman el polígono y que contiene todos los puntos [18]. El Convex Hull de un conjunto finito de píxeles de un objeto en una imagen es el conjunto convexo más pequeño que contiene todos los píxeles (Fig. 1).

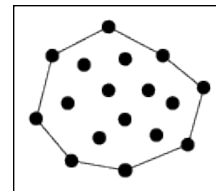


Fig. 1. Conjunto convexo de puntos.

### 2.7 Eliminación de ruido

Ya obtenida la imagen objeto de inspección umbralizada adecuadamente se procede a eliminar el ruido que se presenta, para ello se utilizan técnicas de procesamiento morfológico [18] tales como remoción de objetos pequeños, que elimina los objetos con el menor número de píxeles en conjunto de una imagen binaria, y remoción de objetos de borde que elimina todos los conjuntos de píxeles cuya estructura toca las fronteras de la imagen, útil cuando se sospecha que la información contenida en estos objetos está incompleta [19]. El resultado es una imagen más limpia, con objetos únicamente del tamaño deseado.

## 2.8 Análisis de partículas (estática)

Después de crear una imagen binaria y mejorarla se puede hacer un análisis de partículas que proporciona información como número de objetos, mediciones para determinar la ubicación y características de forma, área en píxeles, entre otras [20], todos estos datos son empleados durante el procesamiento de la imagen y la posterior clasificación de la información.

## 2.9 El prototipo

El prototipo consta de una rampa con inclinación ajustable, donde  $0^\circ$  son una ubicación completamente horizontal. Con inclinaciones muy pequeñas, el flujo de agua con peces podría ser tan bajo que los alevines no desciendan a una velocidad constante. Inclinaciones muy grandes generarían un flujo muy alto, impidiendo que la cámara no sea lo suficientemente rápida para detectarlos. El prototipo inicial consiste entonces en una estructura de acrílico con fondo blanco, rampa inclinable ajustable y soporte para la cámara (Fig. 2).



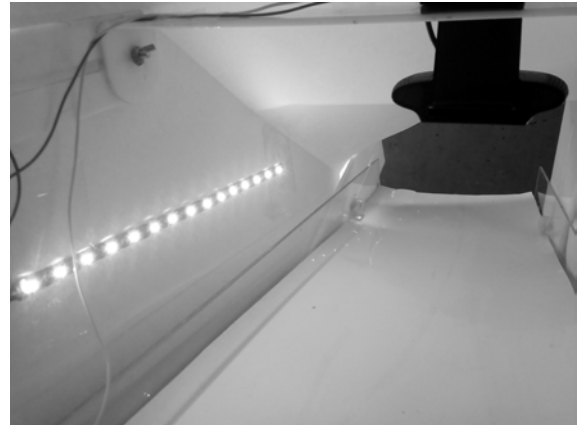
**Fig. 2.** Prototipo en acrílico con rampa de inclinación variable

Algunas especificaciones técnicas del prototipo son:

Dimensiones: 80x40x30cm (largo-ancho-alto)  
 Peso: 2Kg  
 Material: Acrílico y tornillos de hierro  
 Iluminación: Cinta led 12V 15 leds x2 consumo total de 380mA.

La cámara debe ir ubicada a por lo menos 20cm del flujo de alevines, puesto que es una cámara web y su diseño y especificaciones técnicas así lo exigen. Su posicionamiento debe ser perpendicular a la rampa, para tomar la imagen de los alevines lo más precisa posible y sin deformaciones, aunque

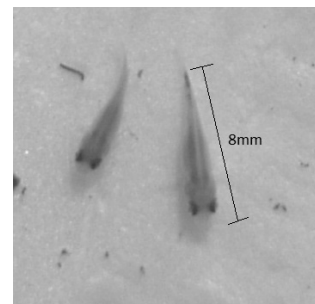
pequeñas variaciones en esta inclinación no afectarían la medición. La ubicación de la iluminación debe ser de tal manera que no refleje los leds en el fondo blanco de la imagen, por lo tanto, están ubicados a los lados de la rampa fuera del foco de la cámara (Fig. 3).



**Fig. 3.** Iluminación interna y ubicación de la cámara

## 3. RESULTADOS Y ANALISIS

Con el fin de realizar el conteo de forma ágil y automática usando técnicas de visión artificial, se partió de la identificación de un alevín, y definición del fondo de la imagen. Se prueba eligiendo color blanco, ya que los alevines tienen un color rojizo que haría buen contraste con el blanco, permitiendo así segmentar la imagen apropiadamente. Los alevines pueden tener un tamaño promedio de 8mm (Fig. 3), pero pueden ser más grandes o más pequeños dependiendo de los días de vida y evolución propia individual. Comúnmente en un lote, se encuentran de diferentes tipos. En el medio que se encuentran, también puede haber pequeñas partículas, como residuos de alimentos, arenas, pequeñas piedras, material orgánico, etc., impurezas que deberían tenerse en cuenta para la realización de la operación de conteo.



**Fig. 3.** Foto de alevines de Tilapia Roja

### 3.1 El algoritmo

El algoritmo de visión artificial es desarrollado con ayuda del asistente de visión de LabVIEW y la metodología utilizada para la creación de este se define con los siguientes pasos:

- Se define una fotografía típica de alevines
- Se implementa un algoritmo de adquisición de imagen en LabVIEW
- Se le aplican los diferentes algoritmos para realizar conteo, definidos en el capítulo 2
- Se validan resultados con otras fotografías
- Se modifica el algoritmo obtenido para conteo en dinámico (video)
- Se prueba y se validan resultados

### 3.2 Aplicación de cada algoritmo a una fotografía

#### Máscara

La máscara se aplica para eliminar alguna zona de la fotografía. En la Fig. 4a no se evidencia aún su utilidad, puesto que la imagen no contiene zonas que deseen eliminarse, sin embargo en las pruebas finales este primer paso será de gran importancia para eliminar bordes de la rampa que la cámara alcanza a tomar. En la figura se conservará únicamente el interior del rectángulo o región de interés (ROI por sus siglas en Inglés).

#### Segmentación por color

Se define un umbral de colores en formato RGB que permita extraer la información necesaria y eliminar el resto. Esta operación resulta ser muy robusta, puesto que los alevines tienen muy buen contraste con el color de fondo, por lo cual los valores ajustados no son muy fijos. Sin embargo, otras partículas oscuras también quedan en la imagen, obligando a generar otros procesos para mejorarla. Los parámetros del algoritmo de umbral de color son:

R: Entre 0 y 226

G: Entre 0 y 217

B: Entre 0 y 107

Nótese que considerando estos rangos se excluye el blanco, cuya combinación es  $R=G=B=255$ . La razón por la cual los valores son inferiores a 255 es porque en fondo no resulta ser un blanco puro. En la Fig.4b se muestra la imagen binaria resultante.

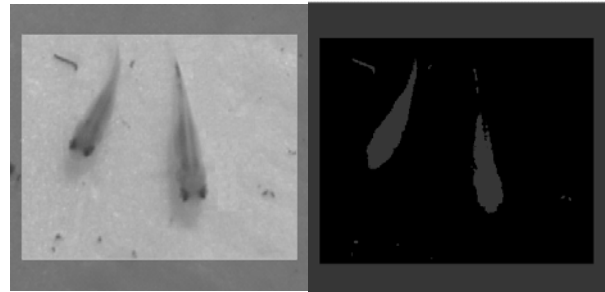


Fig. 4. a) Creación de máscara b) Segmentación por color RGB

#### Remoción de bordes

En el anterior paso, también se generó un recuadro alrededor de los alevines, el cual se creó cuando se hizo la máscara. Su eliminación es muy sencilla, a través de una operación de remoción de bordes.

#### Convex Hull

Esta operación permite que se eliminen concavidades en la morfología de los alevines, haciéndolos más homogéneos, redondos y sólidos. Esto sobretodo es importante si en operaciones posteriores se realizan erosiones o remoción de objetos pequeños, que harían que la imagen se deteriora o desaparezca. Además, esta operación mejora su aspecto para presentarse en una interface de usuario. La Fig. 5a muestra la imagen después de su procesamiento con estas dos últimas operaciones.

#### Remoción de objetos pequeños

La última modificación que se realiza a la imagen es la eliminación de las pequeñas impurezas que quedan alrededor de los alevines. El algoritmo realiza una serie de iteraciones, recorriendo completamente cada posición de la matriz binaria. A mayor cantidad de iteraciones removerá objetos más grandes. Con 5 iteraciones bajo una conectividad 4/8 resulta ser suficiente (Fig. 5b)

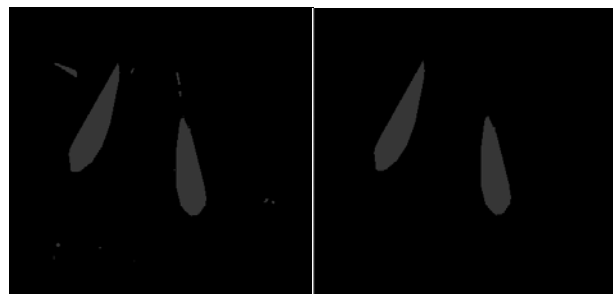
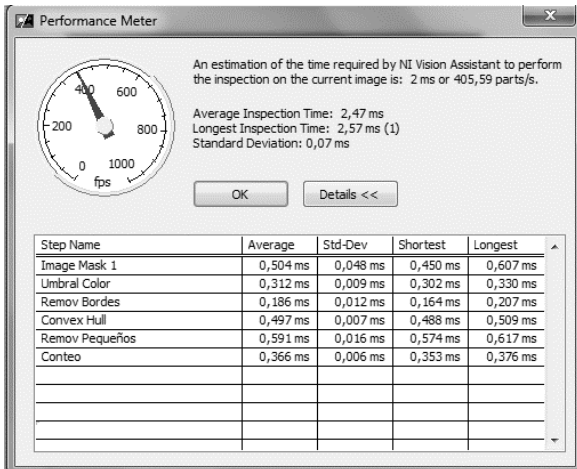


Fig. 5. a) Imagen binaria sin borde y con convex hull b) Remoción de objetos pequeños

**Conteo**

Un último algoritmo de conteo no procesa la imagen, pero sí identifica la cantidad de objetos sólidos que hay en ella. Este permite seleccionar diferentes mediciones como área, centro de masa, coordenadas, perímetro, etc., pero para los alcances de este artículo no son de interés. Para este caso el resultado del conteo es 2.

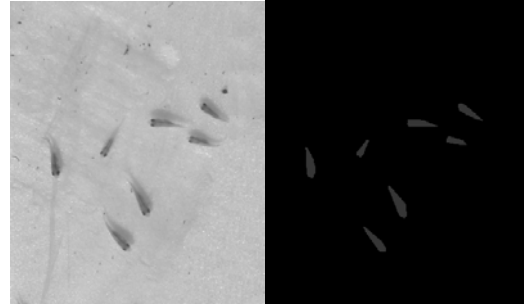
En estos pasos de procesamiento y mejoramiento de la imagen, sin contar con los procesos de adquisición, ni posteriores cálculos, LabVIEW permite calcular su desempeño (Fig. 6). El que más tarda en promedio es el de remover objetos pequeños con una duración de 0.591ms, mientras que el más rápido es el de remover bordes con 0.186ms. El tiempo total promedio es de 2.47ms, permitiendo procesar 405.59 imágenes por segundo.



**Fig. 6.** Medición de desempeño del procesamiento de la imagen

El algoritmo implementado hasta el momento tiene un excelente desempeño, probando con otra imagen de características de iluminación similares

pero con diferente cantidad de alevines se tienen el resultado de la Fig. 7., donde se cuentan 7 alevines.



**Fig. 7.** Validación del algoritmo con imagen de 7 alevines a) Imagen original b) Imagen procesada

**3.3 Adquisición de video**

El siguiente paso se convierte entonces en cambiar el método de adquisición, que la fuente de las imágenes sea video y que el algoritmo tenga un ciclo para hacer la captura de las imágenes.

Se utiliza una cámara web Genius FaceCam 1010, con resolución HD y velocidad máxima de 30fps. Es una cámara de bajo costo cuya viabilidad en este tipo de aplicaciones también se pondrá a prueba.

El algoritmo para la captura de video tiene los siguientes parámetros de configuración:

Device NI-IMAQdx: cam2: Facecam1010  
 Acq type: Continuous Acquisition with inline processing  
 Video Mode: 1280x720 MJPG  
 Speed: 30fps

El diagrama de bloques se muestra en la Fig. 8.

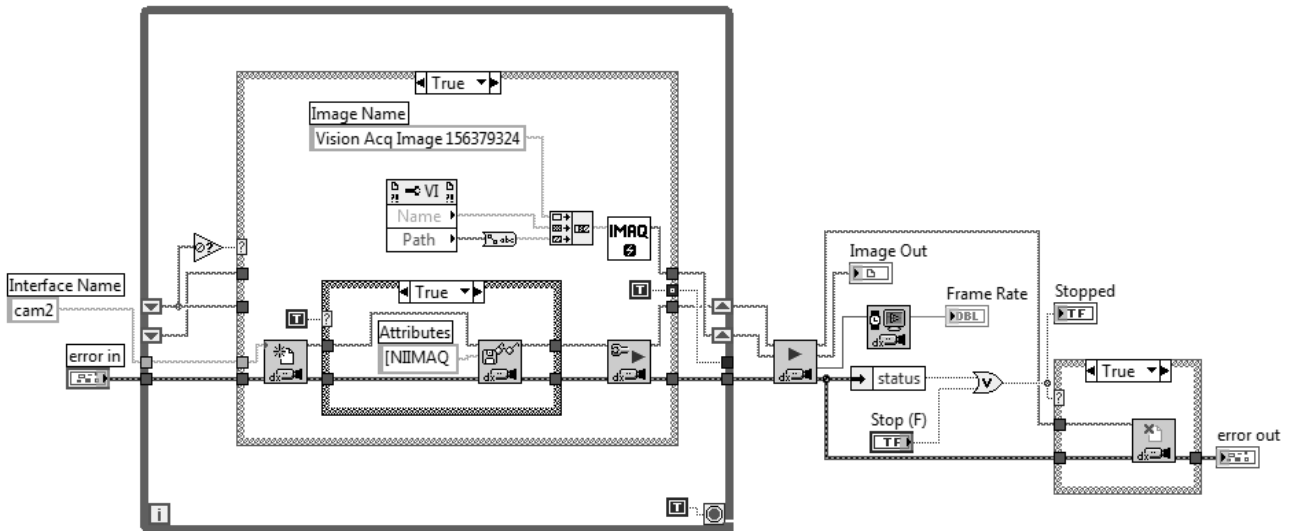


Fig. 8. Diagrama de bloques para la adquisición de video

### 3.4 Conteo en video

El primer inconveniente presentado en la ejecución del conteo de video, es la diferencia en las condiciones de luz presentadas. Por tal motivo se debió encerrar el prototipo dentro de un contenedor, e implementar una iluminación interna. Bajo estas condiciones se logró que las imágenes fueran tomadas siempre bajo las mismas condiciones, independiente de la hora del día, lugar y cualquier otra condición externa.

Con esto solucionado, se procede a desarrollar un algoritmo para el conteo de video. Allí se encuentra una nueva dificultad, puesto que el desarrollo anterior se hizo para una sola imagen, en un solo barrido de un algoritmo secuencial.

Esta dificultad se soluciona fácilmente, si se entiende que un video no es más que una secuencia de imágenes. Por lo tanto todo el algoritmo inicial se inserta dentro de un ciclo *while* con condición de paro por pulsación de un botón. Una vez el algoritmo secuencial está dentro del *while*, se ejecutará una y otra vez, una vez para cada imagen tomada. Dada la velocidad de la cámara, el ciclo se ejecutará con una frecuencia de 30Hz.

Hasta este punto su funcionamiento es sencillo, tiene un indicador numérico en el panel frontal, donde constantemente muestra la cantidad de alevines detectados en cada foto. Este es un número que constantemente está en cambio, puesto que al pasar los alevines frente a la cámara

los detectará, pero una vez éstos se muevan fuera del rango, el indicador volverá a 0. Este indicador numérico se convierte en la principal herramienta para ejecutar el conteo

Se definen entonces 2 variables:

**#Alevines:** Es la cantidad actual de alevines que ve la cámara

**#TotalAlevines:** Es un acumulador que identifica cuántos alevines se han contado.

Se tiene en cuenta la siguiente lógica:

- En condiciones iniciales, ningún alevín es detectado por la cámara. Por lo tanto **#Alevines=0** y **#TotalAlevines=0**.
- Si un alevín ingresa en el rango de visibilidad, **#Alevines** será igual a 1, por lo tanto el acumulador **#TotalAlevines** debe incrementarse en una unidad.
- Si **#Alevines** en un ciclo anterior tenía un valor de 1 por ejemplo, pero en el siguiente ciclo tiene un valor de 3, esto quiere decir que ingresaron 2 alevines al rango de la cámara, por lo tanto el acumulador debe incrementarse en 2 unidades.

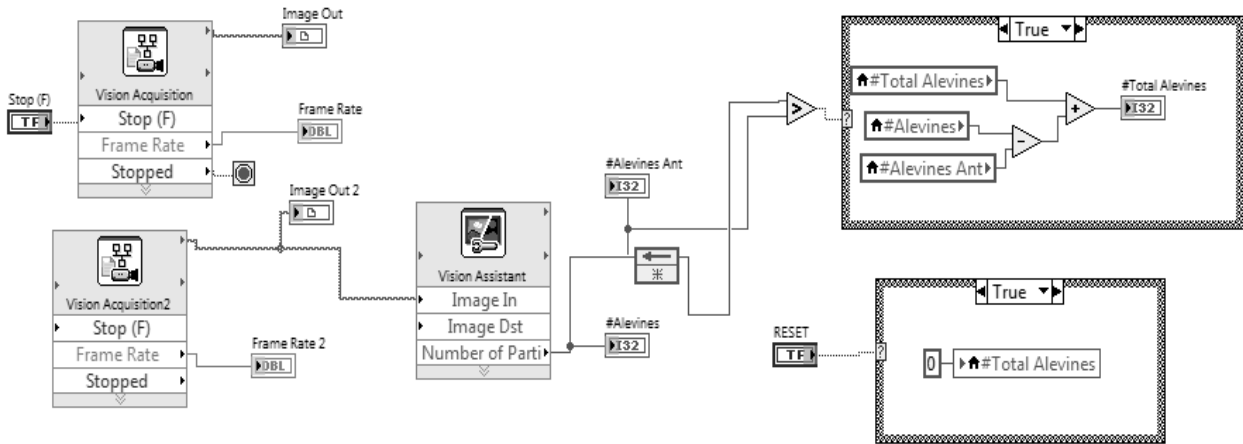
Siendo así, debe definirse una nueva variable que se llame **#AlevinesAnt** y representa el número de alevines que había en la cámara en un ciclo inmediatamente anterior al actual. En programación en LabVIEW esto se hace con un nodo de realimentación (*feedback node*), el cual ante una entrada como **#Alevines** entrega en una salida el



valor que tenía en la iteración anterior, en este caso se almacena en *#AlevinesAnt*.

Para la implementación final se utiliza entonces un comparador, el cual toma el valor de *#Alevines* y lo compara con el valor anterior *#AlevinesAnt*, en

caso que sí lo sea, se ingresa a una estructura *Case* donde se hace la resta  $\#Alevines - \#AlevinesAnt$ , esta diferencia se suma a *#TotalAlevines* (Fig. 9).



**Fig. 9.** Diagrama de bloques del acumulador

Nótese que con esta lógica implementada, un alevín no será contado más de una vez, puesto que solamente afecta al acumulador cuando aparece en la imagen y su posterior aparición no afecta el conteo. Además si aparecen 2 simultáneamente, el algoritmo es capaz de detectarlos.

**3.5 Pruebas**

Se ingresan alevines en la rampa, previa activación del sistema de conteo y reseteo del valor actual. Después de pasar 10 alevines por la cámara, el resultado es superior. La prueba se repite y nuevamente es superior pero diferente, a veces cuenta 15 y en otras ocasiones 30. No hay consistencia en los datos. Se deja habilitado el sistema de conteo, no pasan alevines, pero el acumulador sigue aumentando. Al analizar, las condiciones de luz son diferentes a las cuales para las que fue diseñado el algoritmo, el día es más oscuro o más claro y los resultados seguirán siendo diferentes dependiendo del lugar, hora del día y estado del clima, incluso una sombra puede aumentar el conteo.

Se diseña un cerramiento para la cámara, oscureciendo todo el interior, sólo dejando unas pequeñas aberturas para el canal de agua. Se instala un sistema de iluminación de luz led con las especificaciones técnicas antes descritas.

Se repiten las pruebas, ingresando 10 alevines, el resultado de 5 pruebas se muestra en la Tabla 1.

**Tabla 1.** Prueba inicial de 5 conteos

	1	2	3	4	5
<b>Cantidad</b>	10	10	10	10	10
<b>Conteo</b>	10	8	9	9	8
<b>Error (%)</b>	0	20	10	10	20
<b>ERROR PROMEDIO (%)</b>	<b>12</b>				

En esta ocasión, el conteo nunca es superior, es decir, el sistema sólo está contando peces, pero es frecuente que algunos de los alevines no sean detectados por la cámara. Se tiene un error promedio del 12%.

Se modifica el ángulo de inclinación a 5°, en este caso algunos alevines no recorren de forma homogénea la rampa, quedándose estáticos hasta que otro colisione y los haga continuar su camino. Este tipo de situaciones es indeseable para el algoritmo de conteo. Finalmente se ubica la rampa en una inclinación de 8°, logrando que la velocidad de descenso sea menor y que el flujo sea constante.

Con esta configuración aún hay casos en los que los alevines pasan sin ser detectados. El error promedio en las mediciones es del 8,68%.

**Tabla 2.** Prueba de 20 conteos con 8° de inclinación

<b>PRUEBA</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>
<b>Cantidad</b>	8	9	10	10	10
<b>Conteo</b>	9	8	10	9	8
<b>Error (%)</b>	12,5	11,11	0	10	20
<b>PRUEBA</b>	<b>6</b>	<b>7</b>	<b>8</b>	<b>9</b>	<b>10</b>
<b>Cantidad</b>	10	9	10	10	10
<b>Conteo</b>	8	9	8	9	7
<b>Error (%)</b>	20	0	20	10	30
<b>PRUEBA</b>	<b>11</b>	<b>12</b>	<b>13</b>	<b>14</b>	<b>15</b>
<b>Cantidad</b>	9	10	9	10	10
<b>Conteo</b>	9	8	9	10	9
<b>Error (%)</b>	0	20	0	0	10
<b>PRUEBA</b>	<b>16</b>	<b>17</b>	<b>18</b>	<b>19</b>	<b>20</b>
<b>Cantidad</b>	10	10	10	10	10
<b>Conteo</b>	10	10	9	10	10
<b>Error (%)</b>	0	0	10	0	0
<b>ERROR PROMEDIO (%)</b>			<b>8,68</b>		

Cabe anotar que en algunos casos, el alto flujo de agua aumenta la velocidad de descenso, además de permitir que los alevines se agrumen lo que incrementa las posibilidades de error. Durante las pruebas realizadas fue necesario un entrenamiento del operario para minimizar estos errores generados por la intervención humana.

#### 4. CONCLUSIONES

El algoritmo de procesamiento permite más de 400 partes por segundo, pero la cámara permite 30fps, convirtiéndose en el cuello de botella de procesamiento. El funcionamiento del sistema de conteo mejoraría significativamente implementando una cámara con mayor velocidad, lo que haría más preciso el sistema pero a su vez aumentaría el costo de su implementación.

El conteo arrojó siempre valores iguales o menores a la cantidad conocida, estando presente un error

en las mediciones. El ángulo de 5° resulta ser el más adecuado presentando el menor error en las pruebas realizadas. Si el conteo es superior al esperado, se concluye que son problemas de iluminación. Si el conteo es inferior, se concluye que se debe a que la cámara no pudo detectar algunos alevines, que por su rápido desplazamiento y el tiempo de muestreo del algoritmo no fueron detectados, sin embargo son casos excepcionales y los resultados son aceptables.

Las condiciones de iluminación afectan directamente un algoritmo de visión artificial. Para un equipo móvil como el propuesto en este artículo, se recomienda hacer un cerramiento e instalar iluminación artificial interna con el fin de garantizar en todo momento condiciones estables de luminosidad.

El ángulo de inclinación de la rampa es un factor determinante en el adecuado funcionamiento del conteo, puesto que permitirá modificar la velocidad de desplazamiento de los alevines, haciendo que la cámara pueda detectarlos y garantizar su flujo constante.

Al ser un sistema de visión artificial y conociendo el desarrollo que este tema tiene a nivel mundial, es posible mejorar el algoritmo para extraer más información de los alevines, como talla, color, malformaciones, peso estimado, movilidad y otros factores que son de gran interés para el sector piscícola.

Futuros desarrollos en esta propuesta pueden centrarse en diseñar un equipo a escala industrial, que sea más robusto, resistente, liviano y de fácil transporte e instalación. Además de implementar dispositivos electrónicos embebidos u ordenadores de placa reducida para eliminar la necesidad de contar con un computador como elemento de procesamiento. Adicionalmente, un sistema de suministro de alevines para garantizar el flujo y velocidad adecuados.

#### 5. REFERENCIAS BIBLIOGRÁFICAS

- [1] Angarita P, Maria Rosa, y otros. *Origen y Evolución de la Acuicultura en Colombia*. Bogotá: Observatorio SENA entorno tecnológico de la acuicultura, 2013.
- [2] Merino, María Claudia, Bonilla, Sara Patricia y Bages, Fernando. *Diagnóstico del estado*

- de la Acuicultura en Colombia. Bogotá : Plan Nacional de Desarrollo de la Acuicultura Sostenible en Colombia, 2013.
- [3] Espinal G, Carlos Federico, Martínez Covalada, Héctor J y González Rodríguez , Fredy A. *La cadena de la piscicultura en Colombia*. Bogotá, 2005.
- [4] Charris Slagter, Federico A. *Efectividad de cinco métodos de enumeración de alevines de tilapia (Oreochromis spp.)*. Tegucigalpa, 1998.
- [5] Universidad de los Llanos. Villavicencio. *Ficha técnica para la comercialización de alevinos de Cachama, Yamu, Bocachico y Bagre*. 2012.
- [6] Navarro, Andrés, y otros. *Sistema automático para conteo de peces utilizando sensores infrarrojos y comunicación inalámbrica*: Congreso Internacional en Ingeniería Electrónica, 176-181, 2014.
- [7] NI. Software de Desarrollo de Sistemas NI LabVIEW. Disponible en: <http://www.ni.com> [consultado el 08 de diciembre de 2014]
- [8] González Marcos, Ana, y otros. *Técnicas y Algoritmos Básicos de Visión Artificial*. Universidad de la Rioja, España, 2006.
- [9] Sucar, Enrique y Gómez, Giovanni. *Visión Computacional*. Instituto Nacional de Astrofísica, Óptica y Electrónica. México.
- [10] Marr, David. *Visión*. San Francisco, 1982.
- [11] Snavely, N, Seitz, S y Szeliski, R. *Photo tourism: Exploring photo collections in 3D*. ACM Transactions on Graphics, Boston, 2006.
- [12] Goesele, M, y otros. *Multi-view stereo for community photo collections*. Eleventh International Conference on Computer Vision, Rio de Janeiro. Brasil, 2007.
- [13] Szeliski, Richard. *Computer Vision: Algorithms and Applications*. New York, 2010.
- [14] Sivic, J, Zitnick, C y Szeliski, R. *Finding people in repeated shots of the same scene*. British Machine Vision Conference, Edinburgo. Escocia, 2006.
- [15] Sanabria S, John J y Arhila D, John F. *Detección y análisis de movimiento usando visión artificial*. *Scientia Et Technica.*, 16, 180-188, 2011.
- [16] Platero Dueñas, Carlos. *Inspección automatizada de superficies homogéneas mediante visión artificial con aportaciones al reconocimiento de formas*. Universidad Politécnica de Madrid, Madrid , 1998.
- [17] Gil, P, Torres, F y Ortiz, F G. *Detección de objetos por segmentación multinivel combinada de espacios de color*. España, 2004.
- [18] National Instruments. Disponible en: <http://zone.ni.com> [consultado el 15 de Diciembre de 2014]
- [19] Orellana Lizano, Henry. *Diseño y desarrollo de un algoritmo que permita estimar el tamaño de peces, aplicando visión por computadora, y propuesta para realizar la selección adecuada de dichos peces*. Lima. 2008.
- [20] National Instruments. *IMAQ Vision for LabVIEW*. 2000.