Universidad Católica de Santa María

Facultad de Ciencias e Ingenierías Físicas y Formales

Escuela Profesional de Ingeniería Electrónica

Análisis y monitorización de las características de la espuma en celdas de flotación basados en IoT usando ThingSpeak™ y MATLAB

Tesis presentada por:

Tapia Mayhuire, Gerardo Alexander

ORCID: 0009-0007-1888-7630

Para optar el Título Profesional de Ingeniero Electrónico con Especialidad en Automatización y Control

Asesor:

Dr. Copa Pineda, Juan Carlos

ORCID: 0000-0002-8884-2039

Arequipa – Perú

2026

UCSM-ERP

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

INGENIERIA ELECTRONICA

CON ESPECIALIDAD EN AUTOMATIZACIÓN Y CONTROL

TITULACIÓN CON TESIS

DICTAMEN APROBACIÓN DE BORRADOR

Arequipa, 02 de Febrero del 2026

Dictamen: 014008-C-EPIE-2026

Visto el borrador del expediente 014008, presentado por:

1997225921 - **TAPIA MAYHUIRE GERARDO ALEXANDER**

Titulado:

ANÁLISIS Y MONITORIZACIÓN DE LAS CARACTERÍSTICAS DE LA ESPUMA EN CELDAS DE FLOTACIÓN BASADOS EN IOT USANDO THINGSPEAKTM Y MATLAB

Nuestro dictamen es:

APROBADO

Titulo Profesional/Titulo de Segunda Especialidad/Grado Académico a optar:

INGENIERO ELECTRONICO CON ESPECIALIDAD EN

AUTOMATIZACIÓN Y CONTROL

29364669 - **QUISPE YAUYO JUAN MEDARDO**
DICTAMINADOR

29410027 - **SULLA TORRES RAUL RICARDO**
DICTAMINADOR

29295487 - **URRUTIA ESPINOZA MARIO WILLIAM**
DICTAMINADOR

Análisis y monitorización de las características de la espuma en celdas de flotación basados en IoT usando ThingSpeak TM y MATLAB

INFORME DE ORIGINALIDAD

8%	7%	4%	3%
INDICE DE SIMILITUD	FUENTES DE INTERNET	PUBLICACIONES	TRABAJOS DEL ESTUDIANTE

FUENTES PRIMARIAS

1	repositorio.usm.cl Fuente de Internet	1%
2	iris.uniroma1.it Fuente de Internet	1%
3	qdoc.tips Fuente de Internet	<1%
4	Reza Kohan, Bijan Taheri, Monireh Heshami, Behrouz Maghsodi. "Feasibility Study on the Modification and Improvement of Flotation Circuits at Enrichment Plant 2 of the Sarcheshmeh Copper Complex", Mining, Metallurgy & Exploration, 2020 Publicación	<1%
5	idoc.pub Fuente de Internet	<1%
6	slideplayer.com Fuente de Internet	<1%
7	hdl.handle.net Fuente de Internet	<1%
8	repositorio.uchile.cl Fuente de Internet	<1%
9	patents.google.com Fuente de Internet	<1%
10	Submitted to Universidad Católica de Santa María Trabajo del estudiante	<1%
11	www.slideshare.net Fuente de Internet	<1%
12	"Control predictivo híbrido basado en sensores visuales : teoría y aplicación a flotación de minerales", Pontificia Universidad Catolica de Chile, 2012 Publicación	<1%
13	www.ptolomeo.unam.mx:8080 Fuente de Internet	<1%

DEDICATORIA

Dedico esta tesis a mi querida madre porque me enseñó que, con sacrificio, perseverancia y esfuerzo, se alcanzan las metas pese a las circunstancias y dificultades de la vida.

AGRADECIMIENTOS

"Mas gracias sean dadas a Dios, que nos da la victoria por medio de nuestro Señor Jesucristo" 1 Corintios 15:57

Doy gracias a nuestro Padre celestial, a nuestro Dios y Salvador Jesucristo y al Espíritu Santo porque me dieron la vida, la salvación, la provisión, la ayuda y la guía para obtener la victoria.

Doy gracias a la Universidad Católica de Santa María y a todos los docentes que nos inculcaron valores y pasión por la Ingeniería Electrónica

Doy gracias a mi amada esposa Carolina porque siempre estuvo a mi lado apoyándome en todo.

Doy gracias a mi amada madre Nora y a mi hermana Madeleine porque siempre estuvieron apoyándome y animándome

Doy gracias a mi hijo David y a mis hijas Mercy y Daniela porque fueron el motor para culminar mi carrera.

RESUMEN

En un mundo cada vez más digitalizado, el personal minero enfrenta retos para acceder a las características de la espuma en celdas de flotación mediante dispositivos móviles. Actualmente, son escasas las investigaciones implementadas en plataformas IoT para analizar y monitorear dichas características. Para abordar este problema, esta investigación tuvo como objetivo implementar una arquitectura basada en segmentación de imágenes de la espuma de flotación para medir, analizar y monitorear sus características físicas (tamaño de burbuja y color), estáticas (textura) y dinámicas (estabilidad y velocidad) mediante la integración de un SCADA desarrollado en MATLAB App Designer y un Dashboard IoT basado en ThingSpeak™. Para ello, se capturaron videos de espuma en celdas de flotación de la minera Shougang Hierro Perú S.A.A. Luego, las imágenes se segmentaron mediante técnicas morfológicas, y el algoritmo *marker-controlled watershed* para determinar las áreas y la distribución del tamaño de burbujas (BSD). La estabilidad y la velocidad se determinaron mediante el seguimiento de centroides en frames consecutivos. La matriz *Gray Level Co-occurrence* permitió medir la energía textural y los componentes RGB permitieron obtener la distribución del color de la espuma (FCD). Se implementaron un SCADA y un Dashboard IoT. Los resultados evidenciaron que el método propuesto, basado en marker-controlled watershed, presentó una mayor similitud con el BSD de referencia que el método fast watershed de Wright (1999), con reducciones de 26.14 %, 72.77 %, 47.82 % y 14.60 % en MAE, MSE, RMSE y EMD, respectivamente. Los tiempos computacionales demostraron la viabilidad del monitoreo periódico de las variables de la espuma, con aproximadamente 5 s para su procesamiento en el SCADA y 0.5 s para su transmisión al Dashboard IoT. Se validó la trazabilidad de los datos entre ambas plataformas. Los análisis de tendencias mostraron valores de estabilidad, velocidad, área y textura en rangos de 73 % a 82 %, 10 cm/s a 20 cm/s, 4 cm^2 a 7 cm^2 y 0.13 a 0.16, respectivamente. Se observó que el área de la burbuja presenta generalmente una relación inversa con la velocidad y la estabilidad, y directa con la energía textural. En conclusión, se validó la arquitectura propuesta para el análisis y monitoreo de las características de la espuma, integrada por algoritmos de segmentación, un SCADA *stand-alone* con una tasa de muestreo cercana a 5 s y un Dashboard IoT para dispositivos móviles con una latencia aproximada de 0.5 s, demostrando su viabilidad en el contexto de la Industria 4.0.

Palabras claves: flotación, espuma, IoT

ABSTRACT

In an increasingly digitalized world, mining personnel face challenges in accessing flotation froth characteristics from mobile devices. Currently, research integrating IoT platforms for the analysis, and monitoring of these characteristics remains scarce. To address this issue, this study aimed to implement an image segmentation-based framework for measuring, analyzing, and monitoring the physical (bubble size and color), static (texture), and dynamic (stability and velocity) characteristics of flotation froth through the integration of a SCADA system developed in MATLAB App Designer and an IoT dashboard based on ThingSpeak™. To achieve this, froth videos were captured from flotation cells at the Shougang Hierro Perú mining operation. Subsequently, the images were segmented using morphological techniques, and the marker-controlled watershed algorithm to determine bubble areas and the bubble size distribution (BSD). Stability and velocity were determined through centroid tracking across consecutive frames. The gray level co-occurrence matrix (GLCM) was used to quantify textural energy, while RGB components were employed to obtain the froth color distribution (FCD). Finally, a SCADA system and an IoT dashboard were implemented. The results showed that the proposed marker-controlled watershed-based method achieved greater similarity to the reference BSD than the fast watershed method proposed by Wright (1999), reducing MAE, MSE, RMSE, and EMD by 26.14%, 72.77%, 47.82%, and 14.60%, respectively. Computational times demonstrated the viability of periodic monitoring of froth variables, requiring approximately 5 s for processing within the SCADA system and 0.5 s for transmission to the IoT dashboard. Data traceability between both platforms was validated. Trend analyses revealed values of stability, velocity, area, and texture ranging from 73% to 82%, 10 cm/s to 20 cm/s, 4 cm^2 to 7 cm^2, and 0.13 to 0.16, respectively. It was observed that bubble area generally exhibits an inverse relationship with velocity and stability, and a direct relationship with textural energy. In conclusion, the proposed framework for the analysis, monitoring, and visualization of flotation froth characteristics was validated. The framework integrates segmentation algorithms, a stand-alone SCADA system with a sampling rate close to 5 s, and an IoT dashboard for mobile and tablet devices with an approximate latency of 0.5 s, thereby demonstrating its viability within the context of Industry 4.0.

Keywords: flotation, froth, IoT

ÍNDICE GENERAL

DEDICATORIA .....

AGRADECIMIENTOS.....

RESUMEN .....

ABSTRACT.....

INTRODUCCIÓN 1

CAPÍTULO I: PLANTEAMIENTO DEL PROBLEMA..... 4

1. Identificación del problema..... 4

2. Descripción del problema..... 4

3. Objetivo general..... 5

3.1 Objetivos específicos..... 5

4. Hipótesis 5

5. Variables..... 5

5.1 Variables independientes..... 5

5.2 Variables dependientes 6

6. Alcances de la investigación 6

7. Aporte de la investigación..... 6

8. Antecedentes de la investigación 7

CAPÍTULO II: MARCO TEÓRICO..... 12

1. Proceso de flotación de minerales por espumas 12

2. Sensores para celdas de flotación de espumas 15

3. Máquina de visión para la espuma 16

4. Cámara 17

5. Servidor para la visión artificial 18
6. Luces LED 19
7. Algoritmos de segmentación para medir las características de la espuma 19
7.1 La transformada watershed 19
7.2 Segmentación watershed usando la transformada de distancia 20
7.3 Segmentación basada en watershed usando gradientes 21
7.4 Segmentación watershed controlada por marcadores 22
8. Sistemas de supervisión y monitoreo SCADA 25
9. Internet de las cosas IoT 26
CAPÍTULO III: METODOLOGÍA 30
1. Técnicas 30
2. Procedimientos 30
3. Datos de las variables independientes 30
4. Datos de las variables dependientes 31
5. Instrumentos 31
5.1 Ficha de colección de datos para el análisis de imágenes 31
5.2 Sistema de visión artificial para la caracterización de la espuma de flotación desarrollado en MATLAB 32
6. Campo de verificación 32
7. Ubicación espacial y temporal 33
8. Unidades de estudio 34
9. Estrategia 35
10. Recursos 36
11. Validación del instrumento 36

12. Criterios para el manejo de los resultados 37

CAPÍTULO IV: IMPLEMENTACIÓN DE LOS ALGORITMOS 38

1. Videos o fotos de las burbujas en las celdas de flotación 39

2. Ordenador portátil 42

3. The Image Processing Toolbox de MATLAB® 43

4. Algoritmos de segmentación 44

4.1 Método de segmentación propuesto para medir el área de la burbuja 44

4.1.1 Región de interés (ROI) 47

4.1.2 Ecualización del ROI 47

4.1.3 Binarización 47

4.1.4 Marcador mayor 48

4.1.5 Marcador menor 49

4.1.6 Marcador total 51

4.1.7 Máscara 51

4.1.8 Imposición mínima en la máscara 51

4.1.9 Transformada watershed superpuesta en la ROI 52

4.1.10 Escalamiento del área en cm^2 54

4.1.11 Tendencias para la medición del área 57

4.1.12 Distribución del tamaño de las burbujas (BSD) 58

4.2 Algoritmo para medir la estabilidad de la espuma 59

4.2.1 Criterio para detectar la burbuja estable 59

4.2.2 Sobresegmentación y subsegmentación en la medición de la estabilidad. 65

4.3 Algoritmo para medir la velocidad de la espuma 68

4.3.1 Vector desplazamiento 69

4.3.2 Vector de velocidad media 72

4.3.3 Módulo de velocidad media 73

4.4 Algoritmo para medir el color de la espuma 77

4.5 Algoritmo para medir la textura de la espuma 78

5. Interfaz gráfica SCADA con MATLAB® App Designer 81

5.1 Start buttons, pause, stop y lamps 82

5.2 Numeric edit field para el time period y knob para el H 84

5.3 Edit field para el maximum period, el frame rate y el numbers of frames 85

5.4 Axes para las tendencias de las características de la espuma 85

5.5 Histogramas para las tres componentes de color RGB 87

5.6 Histograma para el BSD 87

5.7 Widgets para las características de la espuma y el progreso del algoritmo 88

5.8 *Edit field* para las características numéricas de la espuma 89

5.9 Axes para el ROI con el área de las burbujas 89

6. Dashboard IoT con ThingSpeak™ 90

7. Diagrama de flujo para el SCADA y el Dashboard IoT 99

CAPÍTULO V: PRUEBAS, RESULTADOS Y DISCUSIONES 103

1. Validación del BSD propuesto 103

1.1 Comparación del BSD de referencia y el BSD propuesto 103

1.2 Resultados de las métricas de error para los BSD 111

1.3 Error absoluto medio (MAE) 112

1.4 Error cuadrático medio (MSE) y raíz del error cuadrático medio (RMSE) 114

1.5 La distancia de Wasserstein (EMD) 118

1.6 Comparación de las métricas de error para los BSD 121

1.7 Discusión de la evaluación de los BSD mediante métricas de error 123

2. Resultados de la clasificación de los BSD por tamaño de burbujas 125

3. Discusión de la clasificación de los BSD 128

4. Resultados de la evaluación de la estabilidad, colapso y coalescencia 128

5. Discusión de los resultados de estabilidad de la espuma 132

5.1 Propuesta de umbrales de alarma para la estabilidad de la espuma 133
6. Resultados de la velocidad de la espuma 134
7. Resultados de la distribución del color de la espuma (FCD) 135
8. Resultados de la energía para la caracterización de la textura de la espuma 137
9. Resultados de los tiempos computacionales de los algoritmos 138
10. Discusión de los tiempos computacionales 143
11. Resultados del SCADA y del Dashboard IoT 144
12. Discusión de los resultados del SCADA y del Dashboard IoT 147
13. Evidencias en video de la investigación 147
CAPÍTULO VI: CONCLUSIONES Y RECOMENDACIONES 154
1. Conclusiones 154
2. Recomendaciones 156
REFERENCIAS BIBLIOGRÁFICAS 159

ÍNDICE DE TABLAS

Tabla 1 Plataformas IoT - closed source 27
Tabla 2 Plataformas IoT - open source 28
Tabla 3 Variables dependientes 31
Tabla 4 Estructura metodológica de la investigación 35
Tabla 5 Presupuesto de los recursos 36
Tabla 6 Características del ordenador portátil 43
Tabla 7 Características físicas, estáticas y dinámicas de la espuma 44
Tabla 8 Propiedades del video con sus valores 46
Tabla 9 Índices de las burbujas estables del 1er y 2do frame 63
Tabla 10 Burbujas mapeadas y estables con sus índices, áreas y centroides 71
Tabla 11 Velocidad de las burbujas estables entre el par de frames 35 y 36 75
Tabla 12 Matriz gray-level co-occurrence GLCM del frame 30 79
Tabla 13 Propiedades de la textura de la espuma para el frame 30 79
Tabla 14 Cálculo de mensajes que se pueden enviar en ThingSpeak 98
Tabla 15 Comparación del error absoluto medio (MAE) 113
Tabla 16 Comparación del error cuadrático medio (MSE) 116
Tabla 17 Comparación de la raíz del error cuadrático medio (RMSE) 117
Tabla 18 Comparación de la distancia de Wasserstein obtenida 121
Tabla 19 Comparación de las métricas de evaluación de los métodos A y B 122
Tabla 20 Comparación de las métricas de evaluación 124
Tabla 21 Características de las métricas empleadas en la evaluación del BSD 125
Tabla 22 Resultados de estabilidad, colapso y coalescencia de las burbujas 131
Tabla 23 Clasificación de alarmas según el nivel de estabilidad de la espuma 133
Tabla 24 Velocidad promedio total VmT obtenida en 1 segundo de video 134
Tabla 25 Tiempo computacional de los algoritmos para el tamaño, estabilidad y velocidad 139

Tabla 26 Cálculo del tiempo computacional código por código para 1 muestra y para 5 muestras 140
Tabla 27 Tiempo computacional de la ejecución completa del código para 1 y 5 muestras en MATLAB® 141
Tabla 28 Tiempo computacional del SCADA implementado con MATLAB App Designer 143
Tabla 29 Comparación de las características de la espuma obtenidas en el SCADA y el Dashboard IoT 144

ÍNDICE DE FIGURAS

Figura 1 Diagrama de flujo general para el procesamiento de minerales 12
Figura 2 Componentes de un proceso de flotación de minerales por espumas 14
Figura 3 Componentes de una máquina de visión para celdas de flotación 17
Figura 4 Arquitectura del servidor para la máquina de visión 18
Figura 5 Arquitectura del flujo de datos en ThingSpeak 29
Figura 6 Ficha de análisis de la segmentación 32
Figura 7 Ubicación de la minera Shougang Hierro Perú S.A.A. 33
Figura 8 a) Molino de bolas, b) Celdas de flotación 34
Figura 9 Esquema conceptual: Análisis y monitorización de las características de la espuma en celdas de flotación basado en IoT usando ThingSpeak™ y MATLAB® 38
Figura 10 a) Agitador y presión de aire de la celda de flotación, b) Presión: 6 psi 40
Figura 11 a) Motor del scraper, b) Espuma junto al scraper 40
Figura 12 a) Transmisor de nivel de espuma, b) HMI con sus variables de control 41
Figura 13 Imágenes de 441 × 441 píxeles denominadas región de interés (ROI) 42
Figura 14 Diagrama de flujo del método propuesto para medir el área de las burbujas 45
Figura 15 a) Espuma a color, b) Espuma en escala de grises, c) ROI de 441 × 441 píxeles 47
Figura 16 Binarización: a) ROI, b) Imagen ecualizada, c) Imagen binaria 48
Figura 17 a) Imagen binaria, b) Algoritmo flood-Fill, c) Marcador mayor 49
Figura 18 a) Ecualización, b) Máxima extendida, H=50, c) Máxima extendida, H=90 50
Figura 19 a) Máxima extendida, H=50, b) Flood-Fill, c) Marcador Menor 50
Figura 20 a) Marcador mayor, b) Marcador menor, c) Marcador total 51

Figura 21 a) La máscara de la imagen ecualizada, b) Marcador total, c) Imposición Mínima del marcador total sobre la máscara 52
Figura 22 a) Watershed en función de la interposición mínima, b) Imagen binaria watershed, c) watershed superpuesta en la ROI 53
Figura 23 Centroides (*) y áreas (píxeles) de las burbujas 54
Figura 24 a) Distancia en cm, b) Izquierda de la burbuja, c) Derecha de la burbuja 55
Figura 25 a) Áreas en píxeles, b) Áreas en cm^2 57
Figura 26 Medición del área de las burbujas durante 7 segundos 58
Figura 27 Distribución del tamaño de las burbujas (BSD) para el frame 35 59
Figura 28 Criterio para detectar la burbuja estable en el 2do frame 60
Figura 29 Interlock para la estabilidad de la burbuja A1 61
Figura 30 Identificación de los índices de las burbujas estables e inestables, frame 1 62
Figura 31 Burbujas estables mapeadas entre el, a) 1er frame, y el b) 2do frame 64
Figura 32 a) Frame 35 con H=50 y b) Frame 36 con área unida, H=95 66
Figura 33 a) Sobresegmentación con H=5 y b) Subsegmentación con H=95 67
Figura 34 a) Centroide del índice 6, frame 35 y b) Centroide del índice 8, frame 36 69
Figura 35 Análisis para el vector desplazamiento $\Delta d_{1,2}$ 70
Figura 36 Medición de la velocidad promedio total en 1 segundo 76
Figura 37 Histograma de la distribución de los colores R G B 78
Figura 38 SCADA: Froth flotation characteristics 81
Figura 39 SCADA: Bottons start, pausa, stop y lamps 82
Figura 40 SCADA: Numeric edit field para el time period y knob para el H 85
Figura 41 SCADA: Maximum period, frame rate y numbers of frames 85
Figura 42 SCADA: Axes para las tendencias de las características de la espuma 86
Figura 43 SCADA: Histogramas para las tres componentes de color RGB 87

Figura 44 SCADA: Histograma para el BSD 88
Figura 45 Widgets para las características de la espuma y los frames 88
Figura 46 SCADA: Edit field para las características numéricas de la espuma 89
Figura 47 SCADA: Axes para el ROI con el área de las burbujas..... 89
Figura 48 Visualización de las características de la espuma con ThingSpeak™..... 90
Figura 49 Características de un mensaje ThingSpeak 91
Figura 50 Características de la Licencia ThingSpeak para estudiantes 92
Figura 51 Campos del canal ThingSpeak para los parámetros de la espuma 93
Figura 52 Capacidad de mensajes comprados y mensajes sobrantes 93
Figura 53 Escritura de las características en campos de un canal ThingSpeak 94
Figura 54 Envió de 10 mensajes con sus campos en 10 intervalos 95
Figura 55 Canal ThingSpeak: Froth Flotation Characteristics with IoT 96
Figura 56 Configuración del canal a modo público 97
Figura 57 Configuración del tiempo del intervalo de actualización a 1 segundo..... 99
Figura 58 Diagrama de flujo para el SCADA y el Dashboard IoT 100
Figura 59 Imagen de referencia (ground truth) 104
Figura 60 Imagen segmentada manualmente 105
Figura 61 Imagen segmentada automáticamente con fast watershed (método A) 106
Figura 62 BSD de la segmentación manual y automática..... 107
Figura 63 Segmentación de la imagen de referencia basada en el método B propuesto..... 108
Figura 64 BSD de la imagen patrón obtenido en el algoritmo de esta investigación 109
Figura 65 Comparación de los BSD de la segmentación manual y automática con el método de segmentación propuesto 110
Figura 66 Clasificación de los BSD promedios en clases A, B, C y D 127
Figura 67 La estabilidad de la espuma es 100 % a) Frame 60, b) Frame 61..... 128

Figura 68 Estabilidad de la espuma del 70 %: a) frame 30; b) frame 31 129
Figura 69 Estabilidad de la espuma del 50 %: a) frame 146; b) frame 147 130
Figura 70 Medición de la estabilidad de la espuma durante 6 segundos 132
Figura 71 Medición de la velocidad de la espuma durante 7 s del video 135
Figura 72 Influencia del tamaño de las burbujas en la distribución del color de la espuma (FCD) 135
Figura 73 Relación entre la energía de textura y el tamaño de las burbujas..... 138
Figura 74 Análisis del tiempo computacional para cinco muestras mediante MATLAB Profiler 142
Figura 75 Tendencias de las características de la espuma con a) MATLAB App Designer y b) ThingSpeak™ 146
Figura 76 SCADA para monitorear las características de la espuma 148
Figura 77 Código QR para el video Froth Flotation Characteristics SCADA - Video 6s CLASE D 148
Figura 78 SCADA en modo oscuro: características de la espuma 149
Figura 79 Código QR: video - Froth Flotation Characteristics SCADA Dark Mode 149
Figura 80 Dashboard IoT para analizar las características de la espuma..... 150
Figura 81 Código QR para el video IoT Dashboard & SCADA..... 151
Figura 82 Dashboard IoT desde un smartphone Samsung A15 para ver el tamaño de las burbujas y la velocidad de la espuma..... 152
Figura 83 Dashboard IoT desde un smartphone Samsung A15 para ver la estabilidad y la textura de la espuma 153

ÍNDICE DE ECUACIONES

Ecuación 1 Factor de escalamiento de píxeles a cm 54

Ecuación 2 Área de la burbuja en cm^2 56

Ecuación 3 Cálculo de la estabilidad en %..... 65

Ecuación 4. Vector desplazamiento 70

Ecuación 5. Vector de velocidad media..... 72

Ecuación 6. Módulo de velocidad media (píxeles/s) 73

Ecuación 7. Módulo de velocidad media (cm/s) 74

Ecuación 8. Velocidad promedio entre un par de frames 75

Ecuación 9. Velocidad promedio total en 1 segundo 77

Ecuación 10. La energía, una de las propiedades de la matriz GLCM. 80

Ecuación 11. Error absoluto medio (MAE) 112

Ecuación 12 Error cuadrático medio (MSE)..... 114

Ecuación 13 Raíz del error cuadrático medio (RMSE) 114

Ecuación 14 Distancia de Wasserstein..... 118

Ecuación 15 Distancia de Wasserstein para histogramas discretos 119

Ecuación 16 Función de distribución acumulada (CDF) de las burbujas reales 119

Ecuación 17 Función de distribución acumulada (CDF) de las burbujas medidas ... 119

ÍNDICE DE ANEXOS

1. Los códigos y la base de datos de las imágenes y videos 164

2. Contactos para solicitar la base de datos..... 164

INTRODUCCIÓN

Tradicionalmente, los operadores de celdas de flotación han dependido de la percepción visual para evaluar la apariencia y forma de las burbujas, así como otras características de la espuma en las celdas de flotación para tomar decisiones de operación como el incremento de reactivos. De esta forma, se puede controlar el tamaño de las burbujas y la velocidad de la espuma con el fin de lograr una buena recuperación del mineral. (Francis, 2001)

Las características de la espuma están relacionadas con la recuperación del mineral y la calidad del concentrado. Existe una fuerte relación entre la recuperación del mineral y el tamaño de las burbujas de la espuma. Por otro lado, la extracción de sólidos (*mass-pull*) o caudal de sólidos que llega al concentrado "se ve afectada por cambios de la estructura y la estabilidad de la espuma, que a su vez se ven afectados por cambios en los parámetros operativos, como la tasa de flujo de aire y la profundidad de la espuma" (Hadler *et al*., 2010, p. 994). Por ende, los controles de procesos ajustan la profundidad de la espuma para lograr la velocidad y el *mass-pull* requerido para una mejor recuperación del mineral (Supomo *et al*., 2008).

Bravo Gálvez (2004) señaló que tanto el exceso como la insuficiencia de aire afectan el proceso de flotación, por lo que es necesario regular adecuadamente el flujo de aire para mantener una espuma estable y favorecer la recuperación de las partículas valiosas. Además, según Massinaei *et al*. (2019), el color de la espuma constituye un indicador del tipo y la cantidad de minerales recuperados en la superficie de la celda de flotación. También señalaron que la energía, como característica textural, aumenta con el flujo de aire y que las imágenes de la espuma parecen ser más uniformes en términos de la distribución del tamaño de las burbujas y de los niveles de gris a mayores flujos de aire. Por otro lado, afirman que un mayor contenido de sólidos en la pulpa favorece el incremento del tamaño de las burbujas y la disminución de la velocidad de la espuma. Farrokhpay (2011) señaló que las espumas más estables suelen caracterizarse por presentar burbujas de menor tamaño y menores eventos de

coalescencia y colapso. Por eso, es importante saber las medidas de las características de la espuma, tales como la distribución del tamaño de las burbujas (BSD), velocidad, estabilidad, color y textura de la espuma, para poder analizar y controlar el proceso de flotación. Sin embargo, las interpretaciones visuales entre operadores sobre las características de las burbujas varían según los turnos de trabajo, ocasionando resultados inconsistentes y deficiencias en la recuperación de minerales. Es por ello que, en los últimos años, la segmentación de imágenes por visión artificial se ha incrementado en la industria minera, tanto en los procesos de molienda para medir el tamaño de las rocas como en el proceso de flotación para medir las características de la espuma.

Debido a la transformación digital de la industria, los administradores de las operaciones mineras enfrentan nuevos retos para acceder a la información, por lo que se necesitan los datos de las características de la espuma y otros procesos en la nube (IIoT/IoT/Industria 4.0) para que puedan hacer uso de este conocimiento con el fin de realizar análisis de datos y brindar mejoras de producción y seguridad en las celdas de flotación. (Jämsä-Jounela, 2019)

Recientemente se están estudiando diferentes soluciones digitales tales como *digital twin, machine vision, process control system, smart equipments, machine learning* (ML) y el *Internet of Things* (IoT) para aplicarlo a la examinación de minerales de la planta de procesos (Nad *et al*., 2022). En esta investigación se propone el uso de algoritmos de segmentación para el análisis de las características de la espuma de flotación mediante un sistema SCADA y un Dashboard IoT.

En el Capítulo I se presenta el planteamiento del problema. A pesar de los avances en visión artificial, son escasas las investigaciones orientadas a la integración de la medición automática de las características de la espuma con plataformas IoT para el almacenamiento, la visualización y el análisis remoto de la información, debido a que estos datos suelen ser de acceso restringido en la industria minera. Para que una planta minera y sus operadores se mantengan a la vanguardia de la Industria 4.0, es

necesario disponer de las características de la espuma en dispositivos móviles y plataformas digitales, permitiendo su almacenamiento como datos históricos y su posterior análisis para la optimización de la producción y la toma de decisiones operativas.

En el Capítulo II se presenta el marco teórico, donde se explica el proceso de flotación de minerales, los sensores y la máquina de visión para espumas en celdas de flotación, así como los algoritmos de segmentación, la transformada *watershed*, el SCADA y las plataformas que se usan con el Internet de las cosas IoT y sus alcances.

En el Capítulo III se presenta la metodología, donde se muestran las técnicas, los procedimientos, la población, la muestra, las variables independientes y dependientes, los instrumentos para determinar la mejor segmentación para medir la distribución del tamaño de las burbujas. También se especifica el campo de verificación, la ubicación espacial y temporal de la investigación.

En el Capítulo IV se presenta la implementación de los algoritmos, donde se muestra el esquema conceptual de la investigación, los videos y las fotos de las celdas de flotación, la segmentación para el área de las burbujas basada en el algoritmo *marker-controlled watershed*, la segmentación para la estabilidad y velocidad de la espuma y la segmentación para el color y la textura de la espuma. Finalmente se muestra la implementación del SCADA con MATLAB App Designer y la implementación del Dashboard IoT con ThingSpeak™

En el Capítulo V se presentan las pruebas, los resultados y la discusión, donde se muestra la validación métrica del BSD. También se presenta la interpretación del BSD por clases, la medición de los tiempos computacionales de los algoritmos y de la escritura en el Dashboard IoT, los resultados de las características de la espuma, la distribución del color de la espuma, las imágenes del SCADA, el Dashboard IoT implementado y los videos en YouTube como evidencia de la investigación. Y en el Capítulo VI se presentan las conclusiones de esta investigación y las recomendaciones para futuros trabajos e investigaciones.

CAPÍTULO I: PLANTEAMIENTO DEL PROBLEMA

1. Identificación del problema

Actualmente, son escasas las investigaciones basadas en plataformas IoT para el análisis de las características de la espuma en celdas de flotación. Asimismo, la falta de acceso a estas variables mediante servicios en la nube limita su visualización y monitoreo remoto.

2. Descripción del problema

Tradicionalmente, los operadores de celdas de flotación han dependido de la percepción visual para evaluar la apariencia y forma de las burbujas de la espuma en las celdas de flotación (Francis, 2001). De acuerdo con el tamaño de las burbujas y otras características en la espuma, los operadores realizan decisiones de operación como incremento de reactivos y manipulación de otras variables para controlar la forma correcta y la velocidad de la espuma con el fin de recuperar el mineral lo mejor posible. Las interpretaciones visuales entre operadores varían según los turnos de trabajo, ocasionando resultados inconsistentes y deficiencias en la recuperación de minerales. Existe una fuerte relación entre la recuperación del mineral y el tamaño de las burbujas de la espuma, el cual es un dato que debe ser analizado constantemente. Sin embargo, hay otras características de la espuma, tales como velocidad, estabilidad, textura y color, que también requieren ser medidas y analizadas para la toma de decisiones en el proceso.

En esta era digital, los administradores de las mineras tienen nuevos retos para acceder a la información en tiempo real. Por esta razón, se requiere la información sobre las características de las burbujas en la nube (IIoT o IoT/Industria 4.0) para hacer analítica y mejorar la producción y seguridad en las celdas de flotación. (Jämsä-Jounela, 2019)

3. Objetivo general

Segmentar las imágenes de la espuma de celdas de flotación capturadas en la minera Shougang Hierro Perú S.A.A en Marcona, Perú para analizar y monitorear las características de la espuma desde un SCADA y una plataforma IoT.

3.1 Objetivos específicos

a) Implementar algoritmos para segmentar las imágenes de la espuma de flotación para luego medir las características físicas (tamaño de la burbuja y color), estáticas (texturas), dinámicas (estabilidad y velocidad) de la espuma.

b) Diseñar una interfaz gráfica (SCADA) usando MATLAB App Designer para analizar y monitorear las características físicas, estáticas y dinámicas de la espuma.

c) Implementar un Dashboard basado en IoT con ThingSpeak™ para analizar y monitorear los datos de las características de la espuma.

4. Hipótesis

La implementación de una arquitectura basada en la segmentación de imágenes de la espuma utilizando MATLAB e integrada a una plataforma IoT mediante ThingSpeak permite analizar y monitorizar de forma remota las características físicas, estáticas y dinámicas de la espuma en celdas de flotación mediante dispositivos móviles.

5. Variables

Los videos e imágenes capturados proviene de la empresa minera Shougang Hierro Perú en Marcona de donde determinaremos nuestras variables.

5.1 Variables independientes

Se capturaron fotos y videos de las celdas de flotación. La variable independiente es la imagen digital de la espuma contenida en la región de interés.

5.2 Variables dependientes

Las variables dependientes son las siguientes características de la espuma de la celda de flotación:

- Tamaño de la burbuja y el BSD.
- Color de la espuma.
- Textura de la espuma.
- Velocidad de la espuma.
- Estabilidad de la espuma.

6. Alcances de la investigación

Esta investigación tiene un enfoque cuantitativo donde se empleará el conocimiento de la visión artificial para implementar algoritmos de segmentación para medir las características físicas, estáticas y dinámicas de la espuma de flotación. La investigación utiliza imágenes y videos de la espuma de celdas de flotación de la minera Shougang Hierro Perú S.A.A. para la segmentación y medición de sus características físicas, estáticas y dinámicas. No se contempla la implementación de una cámara montada en línea para la adquisición continua de imágenes. Se implementará los algoritmos de segmentación usando *The Image Processing Toolbox* de MATLAB® para obtener las características de la espuma. Luego se diseñará una interfaz gráfica de tipo SCADA usando el MATLAB App Designer para mostrar las características de las burbujas. Usando la plataforma de servicios IoT de ThingSpeak™ se analizarán las características de la espuma usando dispositivos móviles.

7. Aporte de la investigación

Son muy escasas las investigaciones sobre las características de la espuma basadas en IoT, ya que el acceso a los datos mineros es restringido. Por lo tanto, se usará ThingSpeak™, una plataforma de IoT basada en la nube, para visualizar y analizar los datos de las características de la espuma desde sus dispositivos móviles.

Para futuras investigaciones, se puede usar estos datos para diseñar estrategias de control de tipo experto o para controlar la velocidad y el tamaño de las burbujas, o estructuras de *machine learning* o redes neuronales para realizar predicciones de recuperación de minerales usando la computación en la nube y el *big data*.

8. Antecedentes de la investigación

En la última década, se han desarrollado varios sistemas para el monitoreo y control de celdas de flotación utilizando máquinas de visión para mejorar la recuperación de minerales. Existen varias compañías internacionales que proveen máquinas de visión para celdas de flotación con el fin de mejorar la recuperación de minerales, por ejemplo: *FrothSense*™ (Outotec), *FrothMaster™* (Outokumpu), *VisioFroth* (Metso), *PlantVision™* (KnowledgeScape), *WipFroth™* (Wipware) y *ECS/FrothVision* (FLSmith). (Lin *et al*., 2008)

No se ha encontrado tesis o artículos nacionales relacionados con la segmentación de las imágenes de la espuma de flotación para el análisis de sus características desde una plataforma IoT. Por otro lado, sí se han encontrado artículos, tesis y patentes a nivel internacional relacionados con la segmentación de las espumas para obtener sus características. Cabe resaltar que los artículos de espuma de flotación basado en IoT son muy escasos debido a que es un tema muy confidencial y cerrado. A continuación, se mencionará las investigaciones más importantes y recientes.

Hosseini *et al*. (2015), en su artículo "Modeling the relationship between froth bubble size and flotation performance using image analysis and neural networks", expresan y modelan mediante redes neuronales la relación entre las condiciones del proceso y el tamaño de la burbuja. El método para determinar el tamaño de la burbuja es a través del algoritmo *watershed* basado en un marcador adaptativo. Los resultados muestran que existe una fuerte correlación entre las condiciones del proceso y el tamaño promedio de las burbujas de la espuma. Concluyen que, aunque los parámetros metalúrgicos pueden ser estimados solamente a partir de la medida del

tamaño promedio de las burbujas, otras características de las burbujas de la espuma tales como velocidad, color y estabilidad deben ser medidas para lograr predicciones más precisas para el desempeño del proceso.

Rendon Montoya *et al*. (2016), en memoria a la patente internacional *Sistema analizador de burbujas en celdas de flotación basado en visión artificial*, patentan un sistema mediante el cual, por medio de una cámara que captura imágenes de las burbujas para segmentarlas en un sistema embebido en tiempo real, se pueden medir los parámetros de la espuma, tales como velocidad, distribución de tamaño, color y estabilidad de las burbujas en una planta de flotación, y enviarlos a un sistema de control por comunicación industrial *Ethernet/IP* para realizar acciones de control de reactivos, tales como flujo de aire y nivel de la celda.

Unzueta Arce (2016), en su tesis *Sistema de medición basado en visión artificial para circuito de flotación rougher de laboratorio*, implementa un sistema de medición de profundidad de espuma y flujo de aire basado en visión artificial. Obtuvo imágenes de la posición del flotador con cámaras web para medir la profundidad de espuma utilizando los algoritmos *template matching* y filtros Haar. Utilizó imágenes de un rotámetro para medir el flujo de aire. Envió estas mediciones a un sistema de control por medio del protocolo DDE. Se logró obtener la medición de la profundidad de la espuma con un error de medición de ± 0,1 cm, con una tasa de muestreo mayor a una medición por segundo, y la medición del flujo de aire con un error de ± 1,5 L/min. La propuesta es viable, pero con un costo computacional elevado, lo que limita el uso de varios sensores paralelamente.

Mesa *et al*. (2016), en su artículo "Textural image classification of foams based on variographic análisis", expresan que hay varias formas de analizar y clasificar la textura de la espuma, tales como el análisis del espectro de la textura, la matriz de co-ocurrencia de nivel de gris, o el análisis de textura de ondículas (wavelets). El objetivo es clasificar y comparar la espuma de aire con agua (*foam*) por medio del algoritmo *variogram*. Se logró clasificar las imágenes de la espuma de aire con agua en función

del espumante usado con una exactitud de 88.9 %. Los resultados sugieren que la textura de la espuma de aire con agua está relacionada con el tipo de espuma, con efectos en su altura.

Massinaei *et al*. (2019), en su artículo "Machine vision based monitoring and analysis of a coal column flotation circuit", implementan una máquina de visión para celdas de flotación que contienen: (a) condiciones de operación (flujo de aire, porcentaje de pulpa de lodo, profundidad de la espuma, dosificación del espumador y colector), (b) características visuales de la espuma (tamaño de las burbujas, velocidad y color), (c) características de la textura en la espuma (energía, entropía, contraste, homogeneidad y correlación), y (d) el rendimiento metalúrgico (recuperación y el concentrado). Estos valores son registrados simultáneamente para analizar la relación que hay entre las características de la espuma, las condiciones del proceso, y los parámetros de rendimiento. El resultado indica que el sistema desarrollado puede ser usado para diagnosticar las condiciones de procesos, y para predecir el rendimiento del proceso en función de las diferentes condiciones de operación.

Zhang *et al*. (2018), en su artículo "Fuzzy association rule based froth surface behavior control in zinc froth flotation", proponen la combinación de la distribución del tamaño de la burbuja (BSD) y la distribución de la velocidad de la espuma. Ante el reto de encontrar el *set point* óptimo en la generación adaptativa para lograr una superficie óptima, es que se introduce el algoritmo apriori difuso para extraer la regla de asociación entre la condición de trabajo actual y el *set point*. Implementan un módulo difuso de inferencia para generar el *set point* óptimo en forma adaptativa bajo ciertas condiciones de trabajo. Se obtuvieron, después de varios experimentos, resultados satisfactorios debido a la robustez del método.

Zezulka *et al*. (2018), en su artículo "Communication systems for Industry 4.0 and the IIoT", analizan la comunicación *open platform communication - unified architecture* (OPC-UA) para interconectar y controlar las interfaces de comunicación con los protocolos de información para cumplir con la Industria 4.0. También se explica

el estado de arte de las comunicaciones abiertas las cuales son necesarias para el desarrollo de la industria del internet de las cosas (IIoT) y los conceptos de la Industria 4.0.

Jämsä-Jounela (2019), en su artículo "Future automation systems in context of process systems and minerals engineering", explica que la megatendencia en el campo de los procesos de automatización son la Industria 4.0, el internet de las cosas de la industria, la computación en la nube y la inteligencia artificial. Asimismo, se explica la arquitectura de la automatización de plantas en el contexto de ingeniería de minerales. Luego, presenta el estado del arte de las herramientas tecnológicas en la industria minera. Se concluye con una discusión sobre nuevas investigaciones de sistemas de procesos motivadas por la Industria 4.0.

Ostadrahimia *et al*. (2020), en su artículo "A new empirical model to calculate bubble size in froth flotation columns", expresan que el tamaño de las burbujas es uno de los parámetros más efectivos en el proceso de flotación y lo suelen medir encontrando el diámetro de la burbuja usando equipos especiales. Esta medición consume un alto tiempo de procesamiento. Se propone un nuevo método para medir la burbuja de forma matemática requiriendo los parámetros tales como retención del gas, velocidad superficial del gas, velocidad superficial del líquido y la velocidad del fluido. Los resultados muestran que los valores estimados en este modelo tienen una exactitud mayor que la de los modelos publicados anteriormente.

Nad *et al*. (2022), en su artículo "Digitalization solutions in the mineral processing industry: The case of GTK Mintec, Finland", estudian las diferentes soluciones digitales tales como digital twin, machine vision, process control system, smart equipments, machine learning (ML), and *Internet of Things* (IoT) para aplicarlo a la examinación de minerales. El objetivo es proporcionar una vista clara de los diferentes aspectos para la digitalización de las plantas de procesamiento de minerales basado en lo aprendido de las plantas GTK Mintec. Debido a la propiedad intelectual y confidencialidad del proyecto, es que se restringen los resultados de la planta de

procesos. Sostienen que el siguiente paso es crear plantas virtuales basados en la planta real por medio de la herramienta *digital twin*. La visión para el futuro es poder simular el proceso para enseñar a las maquinas por medio de ML y IoT a resolver problemas basado en los datos históricos.

De Nardis *et al*. (2022), en su artículo "Internet of things platforms for academic research and development: A critical review", proporcionan un resumen de los tipos de plataformas *closed source* y *open source* enfocándose en cinco criterios: (a) protocolo de comunicación, (b) visualización de datos, (c) procesamiento de datos, (d) integración y (e) seguridad con dispositivos externos.

Cabe recalcar que los artículos sobre el análisis de las características de la espuma desde una aplicación IoT son muy escasos y restringidos. Por ende, se desea realizar esta investigación sobre el análisis y monitoreo de las características de la espuma desde una plataforma IoT usando ThingSpeak™ y MATLAB®.

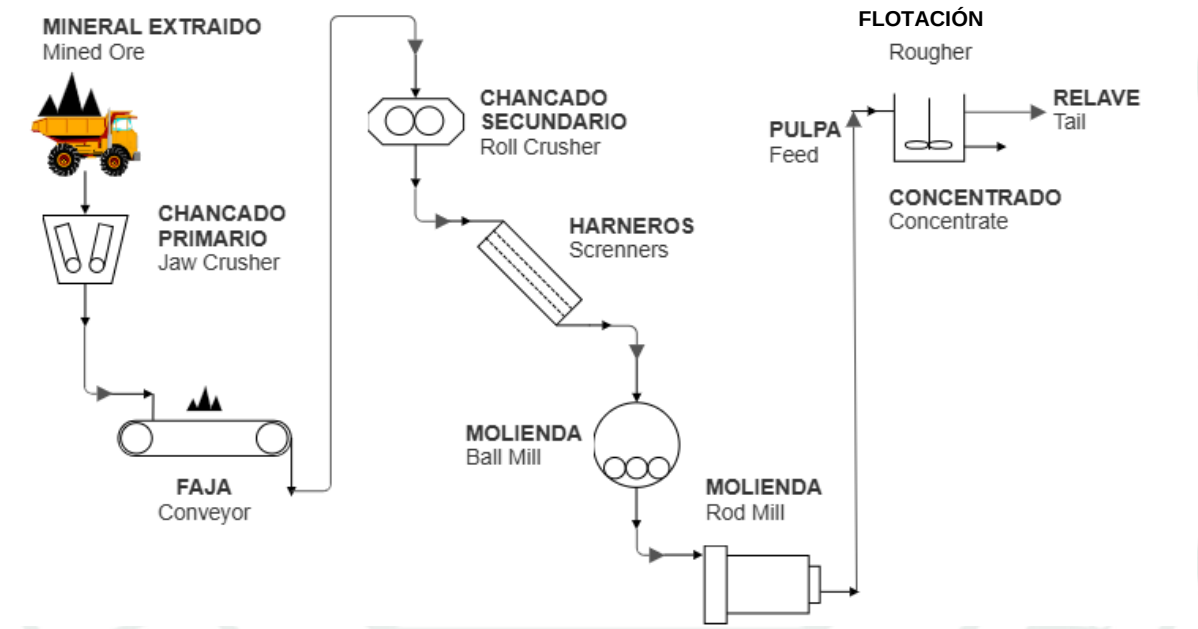
CAPÍTULO II: MARCO TEÓRICO

1. Proceso de flotación de minerales por espumas

El diagrama de flujo de procesamiento de minerales se muestra en la Figura 1, donde se observa que, luego de haber chancado y molido el mineral, se pasa a la flotación para obtener el concentrado de minerales.

Figura 1

Diagrama de flujo general para el procesamiento de minerales



Nota. Elaboración propia. Diagrama de flujo metalúrgico típico que inicia con la extracción del mineral y sigue con las etapas secuenciales de chancado, clasificación, molienda y flotación.

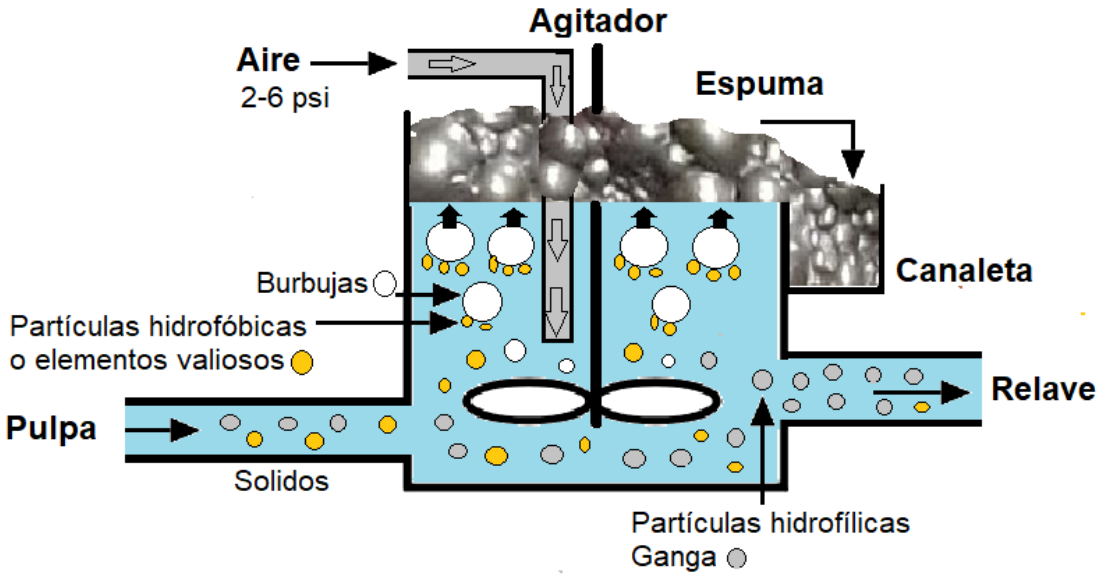
Bravo Gálvez (2004), en su trabajo *Manual de flotación de minerales*, explica la flotación por espumas. Este es un proceso físico–químico proveniente de la concentración de materiales molidos. Primero, la pulpa de mineral tiene que ser tratada químicamente para que los minerales se puedan adherir a las burbujas de aire.

Este proceso permite separar los minerales de la pulpa acuosa aprovechando sus propiedades de afinidad (hidrofílica), o repulsión (hidrofóbica). La propiedad hidrofílica o de afinidad permite que las partículas se mojen para permanecer suspendidas en la pulpa donde finalmente se hunden. Por ejemplo, la mayoría de los minerales estériles de ganga (no deseados), tales como óxidos, sulfatos, silicatos, hierro, carbonatos y otros, presentan esta propiedad hidrofílica. La propiedad hidrofóbica o de repulsión permite que las partículas minerales no se mojen, por lo que pueden adherirse a las burbujas y ascender. Por ejemplo, la plata, el cobre, el oro, el grafito, el carbón, el bituminoso, el talco y otros, presentan esta propiedad hidrofóbica. La adición reactivos químicos a la pulpa acuosa permite hacer más pronunciadas las propiedades hidrofílicas e hidrofóbicas. Por ejemplo, el hierro es originalmente hidrofílico. Sin embargo, para el proceso de flotación, este se vuelve hidrofóbico mediante el uso de reactivos llamados colectores. Se incorpora presión de aire en la parte inferior de la pulpa para generar burbujas de aire con el fin de que las partículas con propiedades hidrofóbicas se adhieren a la burbuja de aire y asciendan hacia la parte superior de la celda de flotación mientras que las partículas hidrofílicas caen al fondo de la celda de flotación. Esta es la forma de separar la parte valiosa o concentrado que se encuentra en las burbujas y el relave que contiene la parte no deseada o estéril al fondo de la celda. En toda celda de flotación intervienen 4 factores importantes que son: (a) la pulpa, (b) los reactivos, (c) la agitación y (d) la presión de aire.

En la Figura 2 se observan los componentes de un proceso de flotación de minerales por espumas.

Figura 2

Componentes de un proceso de flotación de minerales por espumas



Nota. Elaborado a partir de "Fig. 1: Laboratory flotation machine", por V. Kuňa, J. Balberčák, Š. Boháček y V. Ihnát, 2021, *Wood Research*, *66*(2), p. 224. Basado originalmente en *Fundamentals of the theory of flotation*, por N. Petrovskaya, 2007.

La pulpa es una mezcla de mineral molido con agua. Cuando la pulpa contiene partículas muy gruesas mayores a la malla 48, las partículas se asientan en el fondo de la celda. Cuando las partículas son muy finas, menores a la malla 270, la recuperación de los materiales valiosos no va a ser efectiva, ya que se perderían en forma de lamas. Por lo tanto, la pulpa contiene sulfuro valioso, ganga y agua, y debe cumplir con la densidad y el pH correctos. El pH mide de 0 a 14, donde 7 es neutro (ni ácido ni alcalino); es decir, es igual al agua pura.

El aire sirve para agitar la pulpa generando la formación de las burbujas. Los conjuntos de burbujas acompañadas de sulfuros forman las espumas. La espuma se encarga de hacer flotar los elementos valiosos en la superficie de la celda de flotación.

El rango de presión de aire es de 2 a 6 psi. Cuando se usa mucho aire, las burbujas se pueden reventar antes de rebosar por los labios de la celda de flotación o puede ser que rebalse la espuma con la pulpa y la ganga. Si se usa poco aire, no se pueden recuperar los elementos valiosos perdiéndose con el relave. Por lo tanto, si se regula bien el aire, la altura de las compuertas y el dosaje del espumante, se obtendrá una buena espuma. Para lograr una buena recuperación de los sulfuros valiosos, se aplican sustancias químicas llamadas reactivos. Existen tres grupos de reactivos tales como: (a) espumantes, (b) colectores y (c) modificadores. Los reactivos se asocian a los elementos valiosos que más les atraen, pero no a otros. Hay reactivos que combinan bien con el aire, pero hay otros que combinan bien con el agua.

La agitación permite la formación de las burbujas de aire y una mezcla uniforme de los reactivos con los minerales de la pulpa dentro de la celda y evita los asentamientos de los sólidos en la pulpa. Si la agitación es excesiva, las burbujas colapsan y la pulpa se rebalsa en lugar de espuma. Por otro lado, si la agitación es insuficiente, el colchón de la espuma se achica y no alcanza a rebalsar. (Bravo Gálvez, 2004)

2. Sensores para celdas de flotación de espumas

Como señala Sbárbaro y Del Villar (2010), "los sensores de procesamiento de imágenes proveen medidas cuantitativas de ciertas características representativas de la espuma y ofrecen la opción de realizar decisiones de control automático que dependan exclusivamente de ellas" (p. 290).

En los procesos de flotación modernos se requiere un control avanzado. Este consiste en lograr un *closed-loop* por medio de una máquina de visión donde una cámara obtiene las imágenes y un algoritmo segmenta las imágenes para obtener las propiedades de la espuma tales como tamaño de las burbujas, estabilidad, color y velocidad de la espuma por medio de la visión artificial. Estos parámetros o variables de proceso ingresan a un controlador DCS o PLC. Los controladores, mediante reglas,

aplican el control experto para controlar los niveles de reactivos (*reagent*), presión de aire y nivel de la pulpa. Por ejemplo, la data del sensor FrothMaster™, que mide las características de la espuma en las celdas de flotación de la operación minera Cadia Valley en Newcrest, en New South Wales, Australia, es usada para aplicar estrategias de control donde un primer controlador ajusta el nivel de la pulpa (*slurry level*), la presión de aire (*aeration rate*) y la adición de reactivos (*reagent adition*) para controlar el *set point* de la velocidad de la espuma. Por otro lado, el segundo controlador usa la data del analizador de concentrado en línea (*on-stream analyzer*) para manipular el *set point* de la velocidad de la espuma con el objetivo de lograr el *set point* del grado (*grade concentrate*) requerido por los metalurgistas. (Villar & Sbárbaro, 2010)

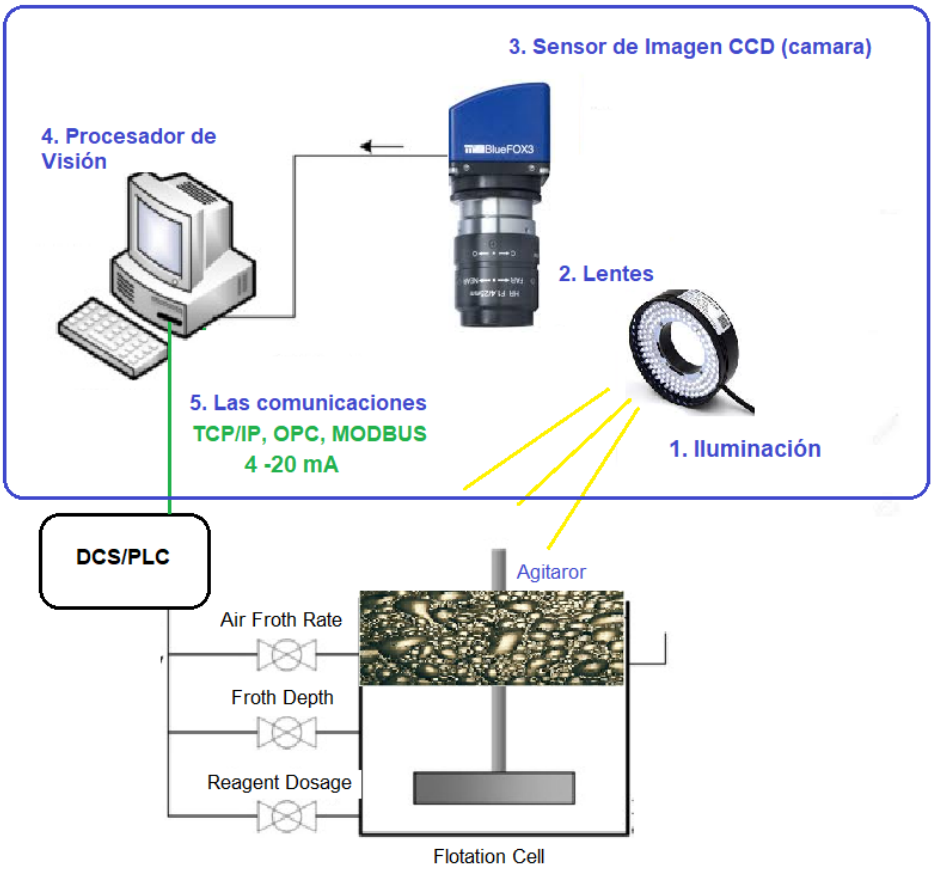
En esta investigación se hará un Dashboard con las características de la espuma, lo cual sería muy beneficioso para que los operadores, desde un dispositivo móvil o tablet, puedan hacer un control manual cuando sea necesario o para analizar y visualizar los datos de las características de la espuma y para tener un registro histórico basado en IoT.

3. Máquina de visión para la espuma

Tradicionalmente, los operadores discriminaban el estado de la espuma para tomar medidas de control como adición de reactivos. Hoy en día, se puede hacer uso de la visión artificial para lograr mejor consistencia en la recuperación de minerales. La máquina de visión, línea azul de la Figura 3, es un instrumento avanzado para aplicarlo en un circuito de flotación. Este es usado para el análisis de las características de la espuma y comprende hardware y software. El objetivo es obtener un reporte de la información sobre la textura, color, tamaño, estabilidad y velocidad de la espuma en un proceso de flotación. Los principales componentes de una máquina de visión para celdas de flotación de espumas están compuestos por: (a) iluminación, (b) lentes, (c) sensor de imagen CCD, (d) PC o procesador de visión y (e) comunicaciones.

Figura 3

Componentes de una máquina de visión para celdas de flotación



Nota. Diagrama técnico de los componentes de la máquina de visión (iluminación, óptica, cámara y procesador) interconectados con el sistema DCS/PLC para la automatización de la celda.

4. Cámara

La cámara es parte del hardware de la máquina de visión, normalmente se usan cámaras CCD de alta velocidad con resolución de 640x480 píxeles de alta calidad con protocolo GigE. Las cámaras de mejor calidad tienen velocidades desde 30 fps con compresión de datos, hasta 100 fps sin compresión de datos y pueden operar hasta 50°C. Es recomendable que tengan interfaz de comunicación Ethernet para la

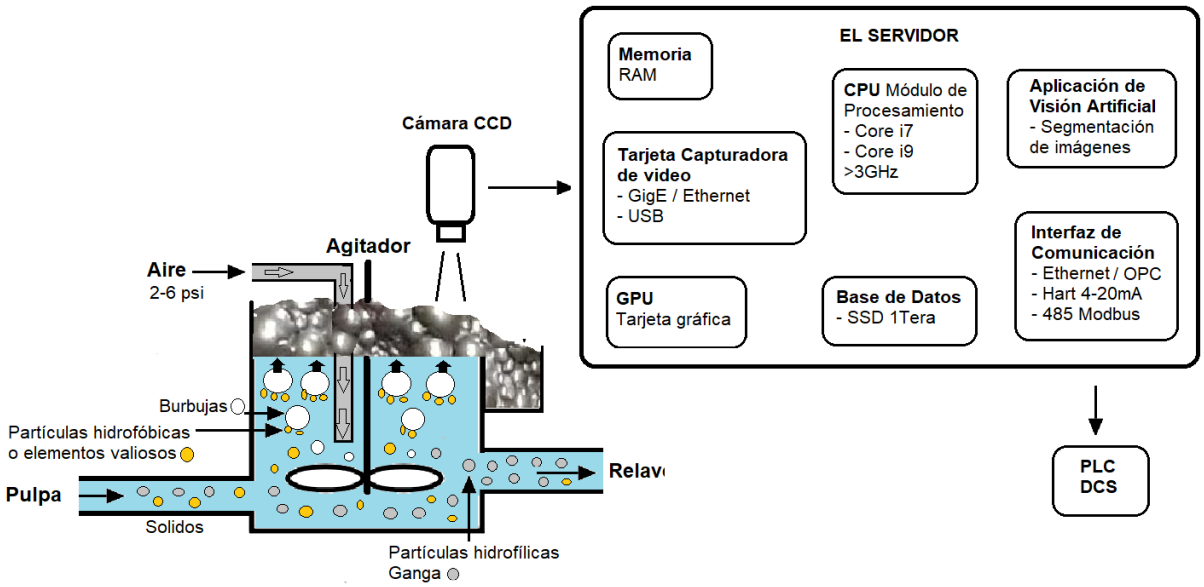
transferencia de datos como las imágenes, y cable de alimentación a través de Ethernet (PoE). Finalmente, la cámara se instala por encima de la celda de flotación.

5. Servidor para la visión artificial

El servidor se muestra en la Figura 4, donde la tarjeta capturadora de video captura las imágenes de la cámara y luego las segmenta a través de una aplicación de visión artificial. Luego, esta aplicación segmenta las imágenes con el fin de determinar el color, tamaño, velocidad y textura de la espuma.

Figura 4

Arquitectura del servidor para la máquina de visión



Nota. Diagrama del sistema que vincula las variables físicas de la celda de flotación con los componentes de hardware (RAM, CPU, GPU, almacenamiento) y software del servidor dedicados al procesamiento de imágenes y la comunicación industrial.

Luego, se almacenan las características de la espuma en una base de datos para obtener tendencias y alarmas. Finalmente, se muestrea las medidas por un puerto de comunicación compatible con protocolos industriales.

6. Luces LED

Las luces LED permiten una mejor vida útil, mayor intensidad y menos consumo. Es de tipo lámpara de anillo de LED difusa. Se requiere aproximadamente 1200 lúmenes para la observación de la burbuja. La cámara requiere de una iluminación uniforme.

7. Algoritmos de segmentación para medir las características de la espuma

Esta es la parte que se desarrollará en esta tesis. Estos algoritmos son el corazón del proceso de análisis de imágenes de la celda de flotación. Estos algoritmos deben medir las características físicas (tamaño de las burbujas y color de la espuma), estáticas (textura) y dinámicas (estabilidad y velocidad) de la espuma. El algoritmo más importante es la transformada *watershed*, y por ende, se explicará cómo funciona con otros ejemplos para luego pasar a implementar los algoritmos de la espuma en el siguiente capítulo.

7.1 La transformada watershed

La segmentación en visión artificial es el proceso de dividir una imagen en varias partes o grupos de píxeles. El objetivo de la segmentación es simplificar o cambiar la representación de una imagen para hacerla más fácil de analizar. Se puede decir que la transformada *watershed* es una técnica morfológica de segmentación basada en regiones que divide toda la imagen en conjuntos conexos. En este ejemplo, se usa la segmentación de cuencas hidrográficas (*watershed*) para separar manzanas. La transformada *watershed* encuentra las cuencas de captación (catchment basin) y las líneas de cresta de cuenca (*drainage lines of a ridge*) de una imagen tratándola como una superficie donde los píxeles de luz son altos y los píxeles oscuros son bajos. Muchas de las funciones para segmentación de imágenes se encuentran en el *Image Processing Toolbox* (IPT) de MATLAB®. (The MathWorks, Inc., 2025a)

Gonzales (2004), en su libro *Digital image processing using MATLAB*, explica la transformada *watershed*. Si se piensa en una imagen con escala de grises como una

superficie topológica donde los valores de *f (x, y)* son interpretados con niveles altos. Si imaginamos que el agua de la lluvia cae en esta superficie, es obvio que el agua se juntaría en las dos áreas etiquetadas como cuenca de drenaje (*catchment basin*). La lluvia sobre la etiqueta de líneas de cresta de la cuenca (*watershed ridge line*) se colectaría igualmente en cualquiera de las cuencas de drenaje.

7.2 Segmentación watershed usando la transformada de distancia

La transformada de distancia es la distancia de cada píxel hacia el píxel de valor no-cero más cercano como se muestra. Para píxeles de valor 1 su transformada de distancia es 0. La función IPT `bwdist(t)` de MATLAB® calcula la transformada de distancia. Para observar la imagen usamos la siguiente función.

```
g = im2bw(f,graythresh(f)
```

La sintaxis para llamar a la función watershed es la siguiente:

```
L = watershed (f)
```

Donde *L* es la etiqueta de la matriz. Los enteros positivos en *L* corresponden a las cuencas de drenaje (*catchment basins*), y los valores ceros indican los píxeles de la línea de cresta de la cuenca (*watershed ridge pixels*). El siguiente paso es complementar la imagen, luego calcular su transformada de distancia y finalmente calcular la transformada watershed del negativo de la transformada de distancia. El código sería el siguiente.

```
gc = -g ;
D = bwdist(gc);
L = watershed(-D)
w = L == 0;
```

Debido a que los valores ceros de los píxeles de *L* representan los píxeles de la línea de cresta de la cuenca (*watershed ridge pixels*), entonces la imagen binaria *w* sirve para mostrar solamente estos píxeles.

Finalmente, para completar la segmentación se aplica una función lógica *and* de la imagen binaria original y del complemento de la imagen *w*.

```
g2 = g & -w
```

Cuando se juntan inapropiadamente, se le llama sobre segmentación, y es un problema común de la segmentación basada en *watershed*. Usando los gradientes, podemos superar este problema.

7.3 Segmentación basada en watershed usando gradientes

La magnitud de gradiente es usada con frecuencia en procesos con escala de grises antes de aplicar la transformada *watershed* para la segmentación. La magnitud de gradiente tiene píxeles de valores altos junto a los bordes de los objetos y píxeles de valores bajos en los otros lugares restantes. Entonces, la transformación *watershed* daría como resultado las líneas de la cresta de la cuenca junto con los bordes de los objetos. Para una imagen *f*, que contiene varias manchas oscuras, se calcula la magnitud de gradiente usando ya sea un filtro lineal o un gradiente morfológico. El código sería el siguiente.

```
h = fspecial('sobel')
fd = double(f)
g = sqrt(imfilter(fd,h,('replicate').^2+imfilter(fd,h,('replicate').^2))
```

Se puede calcular la transformada *watershed* del gradiente para encontrar las líneas de cresta con el siguiente código.

```
L = watershed(g);
wr = L == 0;
```

Cuando no es buena la segmentación, es porque hay muchas líneas de cresta de la cuenca que corresponden al objeto deseado. Este es otro ejemplo de sobresegmentación. Una solución sería suavizar la imagen del gradiente antes de calcular su transformada *watershed*. Para ello, se puede aplicar una apertura-cierre (*close-opening*) donde las últimas líneas sobreponen las líneas de cresta de la cuenca en *wr* como líneas blancas sobre la imagen original como se muestra en el siguiente código.

```
g2 = imclose(imopen(g,ones(3,3)),ones(3,3));
L2 = watershed(g2);
wr = L2 == 0;
f2 = f;
f2 (wr) = 255;
```

7.4 Segmentación watershed controlada por marcadores

Para mejorar las líneas extrañas de cresta, lo cual es difícil determinar cuál cuenca de drenaje (*catchment basin*) está realmente asociada con los objetos deseados. Se puede mejorar este problema usando una segmentación watershed controlada por marcadores (*marker-controlled watershed*). Estos marcadores son entonces usados para modificar la imagen de la gradiente. Empezamos considerando los resultados obtenidos del cálculo de la transformada *watershed* de la gradiente de la imagen sin ningún otro proceso como se muestra en el siguiente código.

```
h = fspecial('sobel')
fd = double(f)
g = sqrt (imfilter(fd,h,('replicate').^ 2+imfilter(fd,h,('replicate').^2)
L = watershed (g);
wr = L == 0;
```

Si el resultado está muy sobresegmentado es debido al gran número de regiones mínimas. La función IPT *imreginmin()* calcula la ubicación de todas las regiones mínimas de la imagen. Su sintaxis es la siguiente.

```
rm = imregionmin(f);
```

Donde:

- *f* = imagen en escala de grises;
- *rm* = imagen binaria cuyos píxeles frontales (f*oreground*) marcan las ubicaciones de las regiones mínimas.

Se puede usar *imregionmin* en la imagen del gradiente para ver por qué la función *watershed* produce tantas cuencas de drenaje (*catchment basins*).

```
rm = imregionlmin(g);
```

La mayoría de las ubicaciones con región mínima son muy superficiales y representan detalles que son irrelevantes para el problema de segmentación. Para eliminar esta región mínima extraña se usa la función IPT *imextendedmin()*, el cual calcula el conjunto de lugares bajos (*low spots*) en la imagen que son más profundos.

```
im = imextendedmin(f,h)
```

Donde:

- *f* = imagen en escala de grises;
- *h* = umbral alto;
- *im* = imagen binaria en la cual los píxeles del plano frontal (foreground) marcan las ubicaciones de las regiones mínimas profundas.

Para obtener el conjunto de marcadores internos, se usa la función *imextendedmin*.

```
im = imextendedmin (f,2)
fim = f;
fim (im) = 175;
```

Las últimas dos líneas superponen las ubicaciones mínimas extendidas como manchas grises sobre la imagen original. Las manchas grises hacen un buen trabajo de marcar los objetos que queremos segmentar.

Como siguiente paso, debemos encontrar marcadores externos que nosotros sabemos que pertenecen al fondo (background). El enfoque que hacemos aquí es marcar el fondo (background) por medio de encontrar los píxeles que están entre las manchas oscuras. Increíblemente, hacemos esto para resolver otro problema del watershed. Específicamente, calculamos la transformada *watershed* de la transformada distancia de los marcadores internos de la imagen *im* como se muestra a continuación.

```
Lim = watershed(bwdist(im));

em = Lim == 0
```

Debido a que estas líneas de cresta están entre el medio de las manchas oscuras marcadas por *im*, estas líneas deberían servir muy bien como nuestros marcadores externos. Teniendo los marcadores externos e internos, ahora podemos usarlos para modificar la gradiente de la imagen usando el procedimiento llamado imposición mínima (*mínimum imposition*). La imposición mínima es una técnica que modifica una imagen en escala de grises para que las regiones mínimas ocurran en los lugares marcados. Otros valores de píxeles son empujados hacia arriba (*pushed up*) como sea necesario para remover todas las otras regiones mínimas. Para implementar esta técnica se usa la función IPT *imimposemin()* y su sintaxis es la siguiente.

```
mp = imimposemin(f,mask)
```

Donde:

- *f* = imagen en escala de grises;
- *mask* = imagen binaria cuyos píxeles frontales (foreground) marcan las ubicaciones deseadas de las regiones mínimas en la salida de imagen *m*;

- *mp* = salida de la imagen que se aplicó la imposición mínima.

Por medio de la región mínima, modificamos la gradiente de la imagen en las ubicaciones de ambos marcadores, externos e internos según la siguiente línea.

```
g2 = imimposemin(g,im|em)
```

Finalmente, se calcula la transformada *watershed* con marcador modificado del gradiente de la imagen para ver el resultado de las líneas de cresta de la cuenca (*watershed ridge líneas*). Las últimas líneas de código superponen las líneas de cresta de la cuenca. (Gonzales *et al*., 2004)

```
L2 = watershed(g2);
      f2 = f;
f2 (L2 == 0) = 255;
```

8. Sistemas de supervisión y monitoreo SCADA

El término SCADA proviene de las siglas en inglés *Supervisory Control and Data Acquisition*. En este tipo de sistemas usualmente existe una computadora, que efectúa tareas de supervisión y gestión de alarmas, así como tratamiento de datos y control de procesos. Un sistema SCADA ofrece la posibilidad de crear paneles de alarma, paneles de datos históricos de la planta descargables en una hoja de cálculo, paneles para cambiar y visualizar los parámetros de control, y la posibilidad de usar gráficos como tanques, motores, etc. Estos elementos están en movimiento simulando el proceso de la planta. En esta investigación se implementará una interfaz gráfica usando MATLAB® App Designer, que permitirá visualizar el SCADA de las características físicas, estáticas y dinámicas de la espuma.

9. Internet de las cosas IoT

Una plataforma IoT permite la recepción de datos provenientes de dispositivos tales como sensores, PLC, aplicaciones como MATLAB®, microcontroladores u otros dispositivos a los cuales se les llama cosas (*things*). Esta plataforma también incluye el almacenamiento, procesamiento, analítica y la visualización de los datos de las cosas (*things*) por medio de la nube (*internet*). Por eso le llaman *Internet of Things* (IoT), es decir, las cosas se pueden ver desde la nube. Por medio del IoT se puede visualizar los gráficos de análisis desde cualquier computadora, tablet o dispositivo móvil que cuente con internet. Existen varias plataformas basadas en IoT. En la Tabla 1, se mostrará un resumen de los tipos de plataformas de tipo *closed source* enfocándose en cinco criterios: a) protocolo de comunicación, b) visualización de datos, c) procesamiento de datos, d) integración y e) seguridad con dispositivos externos. (De Nardis *et al*, 2022)

Tabla 1

Plataformas IoT - closed source

Plataform	Protocols	Data Processing	Data Visualization	Integration	Security
IBMWatson IoT Platform	Device-WISE, Watson IoT SLA(cloud services), MQTT v5, HTTPS	real-time, analytics service add-on; AI and machine learning models, edge analytics	cloud analytics services	API, Node-RED	blockchain, private cloud-ready, link encryption (TLS), authentication (IBM Cloud SSO), identity management (LDAP)
GE PREDIX	OPC-UA Modbus TCP MQTT	Predix Edge Manager	Predix APM Services: browser-based and mobile interface	SCADA Historian IT (CRM, ERP)	Embeded Cybersecurity
AWSIoT	MQTT,HTTP, LoRaWAN	real-time analytics (Rules Engine, Amazon Kinesis, AWSLambda	yes (AWS IoT Dashboard)	REST API	link encryption (TLS), authentication (SigV4, X.509)
Google Cloud IT	MQTT HTTP	BigQuery Cloud Data lab Google machine learning engine	Google cloud Dataflow Cloud Bigtable	API, Cloud Pub/Sub	Secure MQTT
Microsoft Azure IoT	HTTP, HTTPS, MQTT	Azure Stream Analysis manager microservice	Azure stream Analytics Ama	RESTfull Amazon MQ, SNS SQS, SWF, JSON	Authentication and authorization
ThingWorx MDMIoT Platform	MQTT, AMQP, XMPP,CoAP, DDS, WebSockets	predictive analytics (ThingWorx machine learning), real-time analytics (ParStream DB)	ThingWorx SQUEAL	REST API	standards (ISO 27001), identity management (LDAP)

Nota. Adaptado de "Internet of things platforms for academic research and development: A critical review," por L. De Nardis, A. Mohammadpour, G. Caso, A. Usman y M-G. Di Benedetto, 2022, *Applied Sciences*, 12(4), 2172, pp. 6 (https://doi.org/10.3390/app12042172). Copyright 2022 por los autores. CC BY 4.0.

Y ahora en la Tabla 2 se mostrará un resumen de los tipos de plataformas de tipo *open source* enfocándose en los mismos criterios (De Nardis *et al*, 2022).

Tabla 2

Plataformas IoT - open source

Plataform	Protocols	Data Processing	Data Visualization	Integration	Security
FIRWARE	HTTP, MQTT, LoRaWAN, OPC-UA	Generic Enablers	Different Generic Enablers in dashboards	LWM2Mover CoaP, JSON or UltraLight over HTTP/MQTT or OPC-UA, API	Private OAuth2-based authentication, access control
OpenMTC	HTTP, HTTPS, MQTT	Dashboard	no built-in processing capabilities	REST API, JSON serialization	Secure MQTT
ThingsBoard	MQTT, HTTP, CoAP, LwM2M, SNMP	External, through Kafka streams	dashboards and widgets	limited, platform	HTTPS, OAuth 2.0
ThingSpeak	HTTP, MQTT,	HTML JavaScript plugins in a local installation, MATLAB® Analysis Apps in the cloud version	time series visualization and Google Gauges in a local installation, MATLAB® Visualization Apps in the cloud version	JSON, XML, CSV, REST API, ThingHTTP	TLS

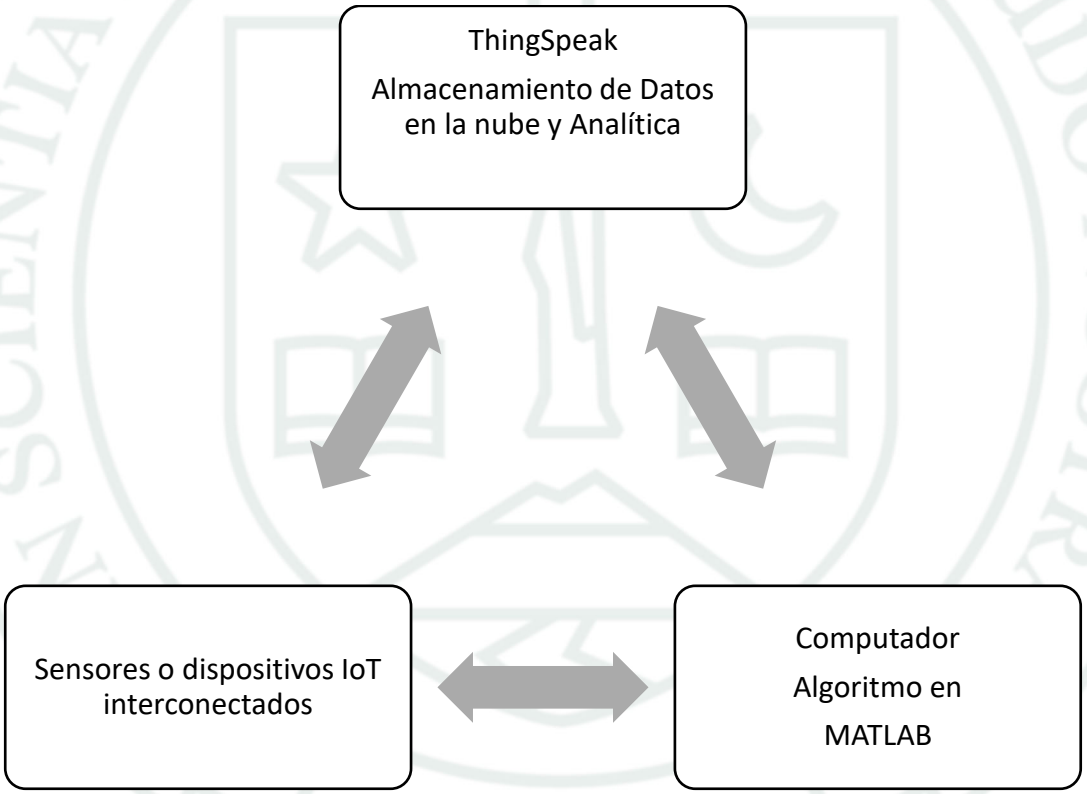
Nota. Adaptado de "Internet of things platforms for academic research and development: A critical review," por L. De Nardis, A. Mohammadpour, G. Caso, A. Usman y M-G. Di Benedetto, 2022, *Applied Sciences*, 12(4), 2172, pp. 7 (https://doi.org/10.3390/app12042172). Copyright 2022 por los autores. CC BY 4.0.

En esta investigación se usará ThingSpeak™ ya que esta plataforma provee servicios de analítica que permiten visualizar y analizar el flujo de datos de las características de la espuma desde la nube. Además, tiene la habilidad de ejecutar códigos basados en MATLAB®. ThingSpeak™ es usado para hacer prototipos que requieren analítica. Como se muestra en la Figura 5, se puede enviar datos desde cualquier dispositivo conectado a internet directamente a la plataforma ThingSpeak™

usando los protocolos REST, API o MQTT. Además, las integraciones de nube a nube con *The Things Network, Senet, the Libelium, Meshlium gateway, y Particle.io* permiten a los datos de los sensores llegar a ThingSpeak™ para los dispositivos móviles con conexiones 4G/3G y LoRaWAN. También se puede almacenar y analizar datos en la nube sin configurar servidores web, y se puede crear alertas por correo electrónico. (The MathWorks, Inc., 2025b)

Figura 5

Arquitectura del flujo de datos en ThingSpeak



Nota. Adaptado de *Sign In - ThingSpeak IoT*, por The MathWorks, Inc., 2026. https://thingspeak.mathworks.com/login

CAPÍTULO III: METODOLOGÍA

1. Técnicas

La técnica empleada en la presente investigación fue la visión artificial basada en procesamiento digital de imágenes. Mediante esta técnica se analizaron imágenes y videos de la espuma de flotación para obtener información relacionada con sus características físicas, estáticas y dinámicas. El análisis permitió caracterizar la espuma a partir de la información contenida en las imágenes y generar indicadores útiles para su monitoreo y evaluación.

2. Procedimientos

Primero, se implementaron los algoritmos en el servidor de visión artificial utilizando el Image Processing Toolbox de MATLAB® para obtener información sobre las propiedades de las burbujas que conforman la espuma, tales como tamaño, cantidad, color, estabilidad y velocidad. En segundo lugar, se utilizó MATLAB App Designer como interfaz gráfica tipo SCADA para visualizar las características obtenidas de la espuma. Finalmente, se empleó la plataforma IoT ThingSpeak™ para el almacenamiento, análisis y visualización de las características de la espuma a través de la nube.

3. Datos de las variables independientes

La variable independiente corresponde a la imagen digital de la espuma contenida en la región de interés (ROI). El ROI consiste en una imagen preprocesada en escala de grises, con una resolución de 441 × 441 píxeles, obtenida a partir de frames de videos capturados en las celdas de flotación de la empresa Shougang Hierro Perú S.A.A., ubicada en el distrito de Marcona, provincia de Nazca, departamento de Ica, Perú.

4. Datos de las variables dependientes

Las propiedades de la espuma medidas en la presente investigación constituyen las variables dependientes de la tesis, tal como se muestra en la Tabla 3. Estas variables se obtienen mediante la segmentación y el análisis de imágenes provenientes de videos en formato MP4 utilizando MATLAB®.

Tabla 3

Variables dependientes

Variables	Unidades
Color de la burbuja	(RGB)
Tamaño de la burbuja	(píxeles o cm^2)
Cantidad de burbuja	(burbujas por frame)
Distribución de tamaño de burbuja (BSD)	Burbujas por intervalo de tamaño
Velocidad de la espuma	(cm/s)
Textura (energía)	0 a 1
Estabilidad de la espuma	%

5. Instrumentos

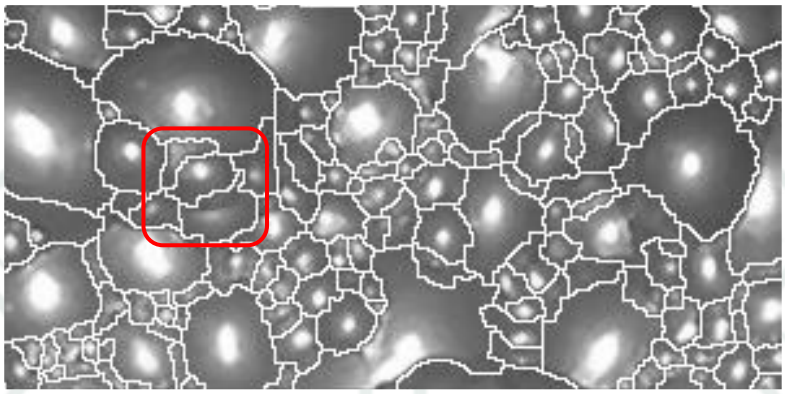
Para la recolección, procesamiento y análisis de los datos se utilizaron los siguientes instrumentos:

5.1 Ficha de colección de datos para el análisis de imágenes

Esta ficha se empleó para registrar los resultados obtenidos durante el desarrollo y evaluación de los algoritmos de segmentación, permitiendo documentar los parámetros utilizados y el desempeño alcanzado en cada prueba, como se muestra en la Figura 6.

Figura 6

Ficha de análisis de la segmentación



Nota. Resultado de la segmentación de burbujas con la función `imextendedmax`, utilizando el parámetro H = 50 para evitar sobresegmentación y subsegmentación.

5.2 Sistema de visión artificial para la caracterización de la espuma de flotación desarrollado en MATLAB

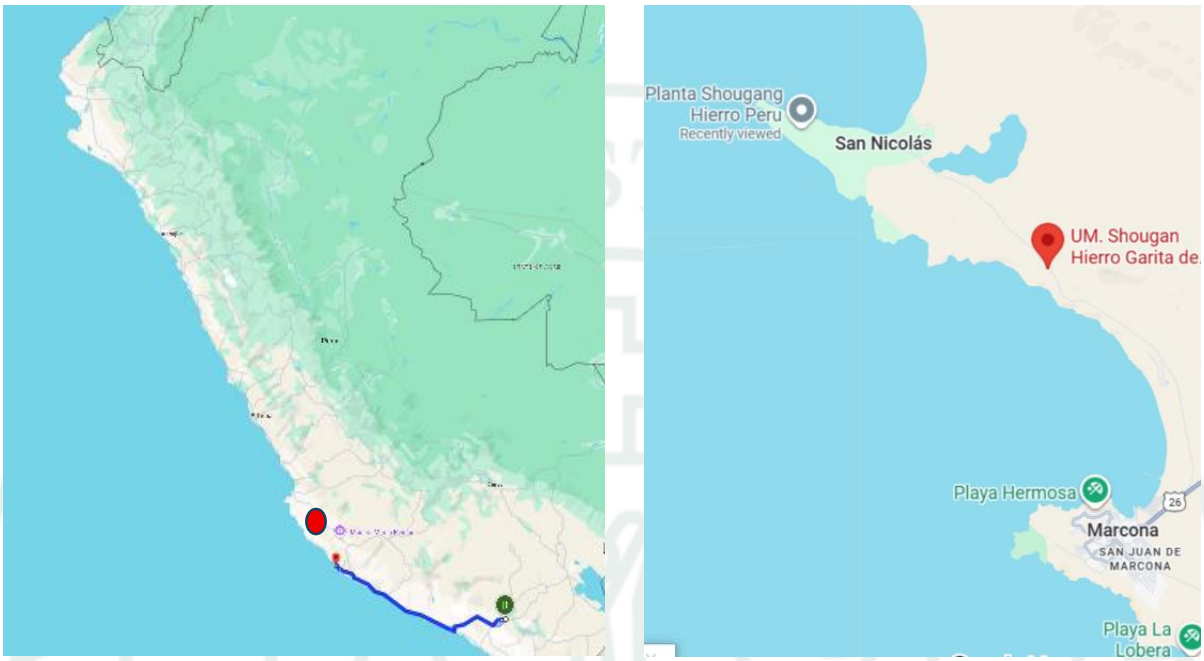
Se desarrolló un sistema de visión artificial utilizando el *Image Processing Toolbox* de MATLAB® para segmentar las imágenes de la espuma y medir sus propiedades. El sistema permitió obtener características físicas (tamaño de la burbuja y color de la espuma), estáticas (textura representada mediante la energía) y dinámicas (estabilidad y velocidad de la espuma).

6. Campo de verificación

Para la presente investigación, las muestras fueron obtenidas en la empresa Shougang Hierro Perú S.A.A., ubicada en el distrito de Marcona, provincia de Nazca, departamento de Ica, Perú. En la Figura 7 se muestra la ubicación de la empresa respecto a la ciudad de Arequipa.

Figura 7

Ubicación de la minera Shougang Hierro Perú S.A.A.



7. Ubicación espacial y temporal

La empresa Shougang Hierro Perú S.A.A. cuenta con diversas áreas de proceso. Para la obtención de las muestras, se autorizó el ingreso a la planta concentradora durante un período de dos horas. En dicha visita se accedió al área donde se encuentran los molinos de bolas y las celdas de flotación, como se muestra en la Figura Figura 8. En esta última área se registraron las muestras de espuma mediante videos y fotografías.

Posteriormente, las muestras obtenidas fueron trasladadas a la ciudad de Arequipa para su procesamiento y análisis. La implementación de los algoritmos de visión artificial se realizó utilizando el *Image Processing Toolbox* de MATLAB® en una computadora portátil.

Figura 8

a) Molino de bolas, b) Celdas de flotación



a)



b)

Nota. Fotografías tomadas durante la visita de campo a la planta de la minera Shougang Hierro Perú S.A.A., mostrando la etapa de molienda y flotación.

8. Unidades de estudio

La población está constituida por las celdas de flotación de la empresa Shougang Hierro Perú S.A.A. Las muestras corresponden a las imágenes y videos de la espuma obtenidos en dichas celdas durante la visita de campo. La unidad de estudio es la Región de Interés (ROI), definida como una imagen preprocesada en escala de grises obtenida a partir de un frame de video. La variable independiente corresponde a la imagen digital de la espuma contenida en la región de interés (ROI). Sobre esta variable se aplican los algoritmos de visión artificial para obtener las variables dependientes asociadas a las propiedades de la espuma, tales como el color, el tamaño de burbuja, la velocidad, la textura y la estabilidad de la espuma.

La Tabla 4 presenta la estructura metodológica de la investigación y resume la relación entre la población, muestra, unidad de estudio, variable independiente, procesamiento y variables dependientes. De manera conceptual, esta relación puede expresarse como $Y = f(X)$, donde X representa la imagen digital de la espuma contenida

en la región de interés (ROI), f representa el procesamiento realizado mediante algoritmos de visión artificial y Y corresponde a las variables dependientes obtenidas.

Tabla 4

Estructura metodológica de la investigación

Elemento metodológico	Descripción
• Población	• Celdas de flotación
• Muestra	• Imágenes y videos de la espuma
• Unidad de estudio	• Región de interés (ROI) (441 × 441 píxeles)
• Variable independiente X	• Imagen digital en pixeles de la espuma contenida en la región de interés (ROI)
• Procesamiento $f(\cdot)$	• Algoritmos de visión artificial
• Variables dependientes Y	• Color, tamaño de burbuja, distribución de tamaño de burbuja (BSD), velocidad, textura y estabilidad.

*Nota***.** La unidad de estudio corresponde al ROI. La relación entre las variables de estudio se expresa como $Y = f(X)$

9. Estrategia

La estrategia de investigación consistió en utilizar técnicas de visión artificial para extraer información relevante de la espuma de flotación a partir de imágenes y videos. Las características obtenidas fueron procesadas, almacenadas y visualizadas mediante herramientas computacionales, permitiendo su análisis a través de un sistema SCADA y un Dashboard IoT. Esta estrategia facilitó la caracterización de las propiedades físicas, estáticas y dinámicas de la espuma.

10. Recursos

La Tabla 5 presenta el presupuesto estimado de los recursos utilizados en la investigación.

Tabla 5

Presupuesto de los recursos

Ítem	Descripción	Precio USD
1	Programa MATLAB® R2020b, student license Incluye Image Processing Toolbox	55.00
2	ThingSpeak™: 1 Unidad. License: 41284718	45.00
3	1 Laptop - Asus Portátil R518UQ-DS54T VivoBook con tapa, 15,6" con procesador Intel(R) Core(TM) i7-7500U CPU @ 2.70GHz 2.90 GHz	527.00
2	Gastos de viaje a Marcona	150.00
	TOTAL	USD 1309.00

11. Validación del instrumento

La validación del algoritmo de segmentación se realizó mediante la comparación de las distribuciones de tamaño de burbujas (BSD) obtenidas por los distintos métodos de segmentación con el BSD de referencia basado en la segmentación manual de Wright (1999). Para ello se emplearon las métricas MAE, MSE, RMSE y EMD, las cuales permitieron evaluar la similitud con el BSD de referencia. Los resultados permitieron verificar la capacidad del algoritmo propuesto para reproducir adecuadamente el BSD de referencia y proporcionar mediciones confiables del número de burbujas y de sus áreas. Posteriormente, la información obtenida fue empleada para el cálculo de la estabilidad y velocidad de la espuma.

12. Criterios para el manejo de los resultados

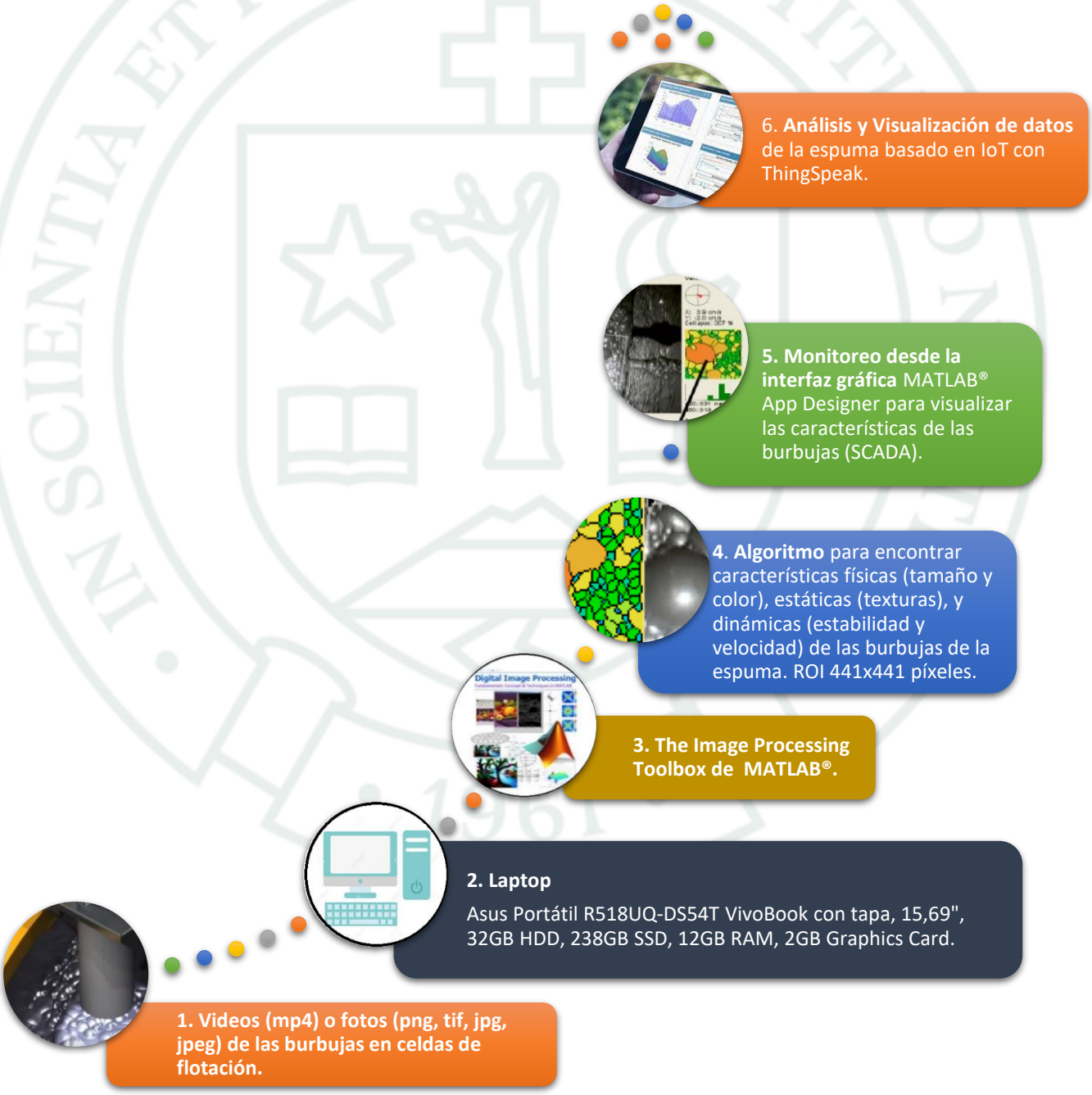
Los resultados obtenidos mediante el sistema de visión artificial fueron organizados y presentados a través de tablas, histogramas, gráficos de tendencias e indicadores visuales, tales como gauges y *displays* numéricos, para facilitar su análisis e interpretación. Las características de la espuma, tales como el tamaño de burbuja, distribución de tamaño de burbuja (BSD), estabilidad, velocidad, color y textura, fueron evaluadas mediante el análisis de tendencias y la comparación de las mediciones obtenidas. Asimismo, la información generada fue visualizada mediante una interfaz SCADA y un Dashboard IoT, permitiendo supervisar el comportamiento de las variables medidas y verificar el desempeño de los algoritmos implementados.

CAPÍTULO IV: IMPLEMENTACIÓN DE LOS ALGORITMOS

En la Figura 9, se muestra el esquema conceptual de la tesis.

Figura 9

Esquema conceptual: Análisis y monitorización de las características de la espuma en celdas de flotación basado en IoT usando ThingSpeak™ y MATLAB®



1. Videos o fotos de las burbujas en las celdas de flotación

Se obtuvieron las muestras en las celdas de flotación de la minera Shougang Hierro Perú S.A.A. Es importante analizar el proceso y los equipos más importantes de la celda de flotación para poder entender la dinámica de la espuma y la recuperación del mineral. También es importante entender que existen varios tipos de celdas de flotación. En algunos procesos de flotación, el oro, la plata y el cobre se adhieren a las burbujas. Para el caso de la minera Shougang, el mineral deseado es el hierro. Entonces el azufre se debe adherir a las burbujas mediante químicos y subir a la superficie de la celda de flotación mientras que el hierro se debe hundir de forma natural debido a sus propiedades magnéticas. Entonces lo que hay en las burbujas es azufre y no hierro. Nuestra investigación no se basa en la recuperación del mineral deseado, sino en encontrar las características de la espuma mediante algoritmos de segmentación y analizarlas en la nube con IoT. Este algoritmo puede trabajar para cualquier tipo de celdas para el análisis del material que se encuentra en la espuma. El color del azufre es medio amarillento, pero como está mezclado con otros químicos tiende a ser de color gris.

Se observó que encima de la celda de flotación se encuentra una válvula de control de aire para insertar una presión de 3 a 7 psi que se usa para formar las burbujas. Si se envía mucha presión, las burbujas colapsan y si la presión es muy baja ya no se forman muchas burbujas y el azufre se cae al fondo y se mezcla con el hierro. Para evitar que el azufre se caiga al fondo hay un motor que mezcla la pulpa como se muestra en la Figura 10.

Figura 10

a) Agitador y presión de aire de la celda de flotación, b) Presión: 6 psi



a) b)

Nota. Fotografías tomadas durante la visita de campo a la planta de la minera Shougang Hierro Perú S.A.A.

Un motor hace girar un raspador (*scraper*) para retirar la espuma con restos de azufre de las celdas, como se muestra en la Figura 11.

Figura 11

a) Motor del scraper, b) Espuma junto al scraper



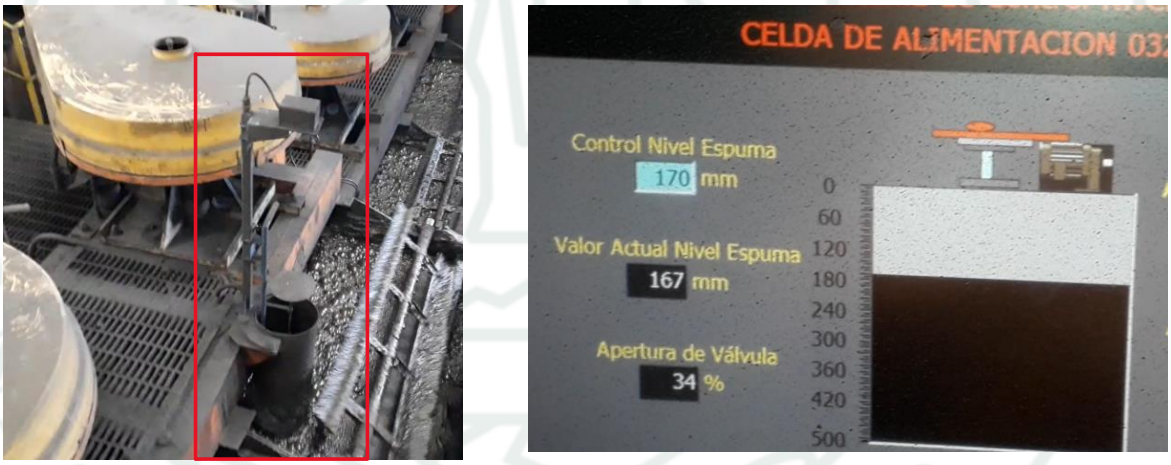
a) b)

Nota. Fotografías tomadas durante la visita de campo a la planta de la minera Shougang Hierro Perú S.A.A.

Cada celda trabaja con un control de nivel de espuma y se puede controlar y supervisar sus parámetros de control desde un HMI de la marca Siemens, como se muestra en la Figura 12 b). Podemos interpretar que el nivel de espuma deseado o *set point* (SP) es igual a 170 mm y que el transmisor de nivel de espuma es la variable de proceso (PV) que nos mide un valor actual de 167 mm, como se muestra en la Figura 12 a). Se observa que el control de nivel tiene muy bajo error, $e = SP - PV = 3\ mm$, donde e es el error, por lo que está controlando adecuadamente. Por ende, el controlador (CV) ordena la apertura de la válvula a un 34 % para que entre poco concentrado a la celda y no se rebalse el tanque.

Figura 12

a) Transmisor de nivel de espuma, b) HMI con sus variables de control



a) b)

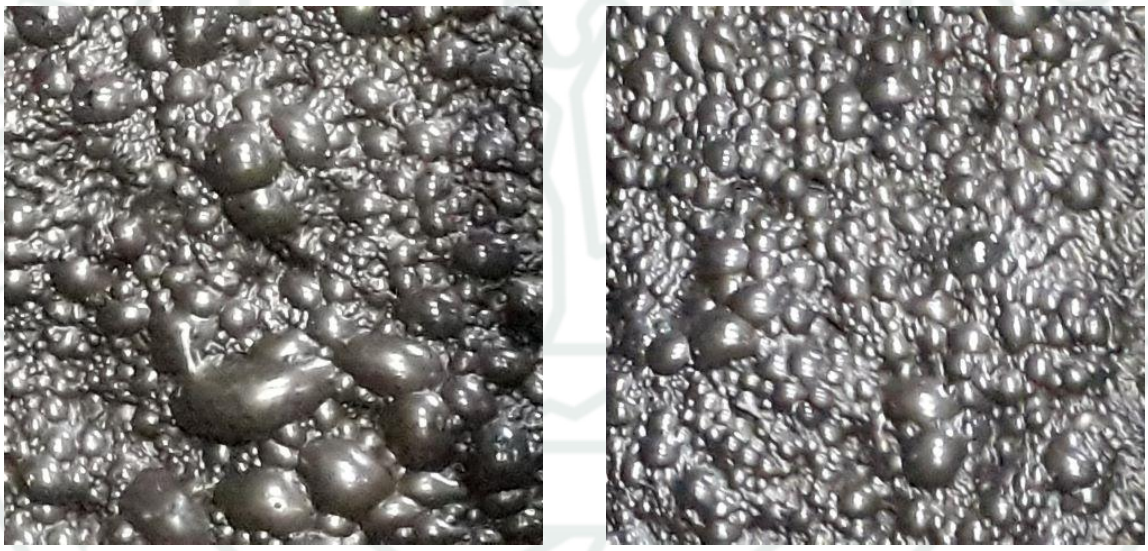
Nota. Fotografías tomadas durante la visita de campo a la planta de la minera Shougang Hierro Perú S.A.A.

Los videos de la espuma capturados con la cámara de un celular Samsung J6 con resolución de 1280 × 720 píxeles constituyen las muestras de la investigación.

La variable independiente corresponde a la imagen digital de la espuma contenida en la región de interés (ROI). Esta es la imagen preprocesada del video en escala de grises con una resolución de 441 × 441 píxeles, como se ve en la Figura 13. Su resolución es baja con el fin de reducir el tiempo de procesamiento cuando se haga la segmentación.

Figura 13

Imágenes de 441 × 441 píxeles denominadas región de interés (ROI)



2. Ordenador portátil

Se usó una laptop con las siguientes características y aplicaciones para este proyecto, como se muestran en la Tabla 6.

Tabla 6

Características del ordenador portátil

Nombre	Descripción
Marca	Asus
Modelo	R518UQ-DS54T VivoBook con
Procesador	Core™ i7
Tarjeta de video	Intel HD Graphics
Memoria RAM	16GB DDR4
Almacenamiento	1TB (HDD)
Sistema perativo	Windows 10
Device Name	DESKTOP-8MTFF13
Processor	Intel(R) Core(TM) i7-7500U CPU @ 2.70GHz, 2.90 GHz
Installed RAM	12.0 GB (11.9 GB usable)
Storage	932 GB HDD TOSHIBA HDWL110, 238 GB SSD
Graphics Card	HFS256G39TND
Device ID	NVIDIA GeForce 940MX (2 GB), Intel(R) HD Graphics 620
Product ID	(128 MB)
System Type	AB89ECCA-2C61-4197-ABE8-A5BEE4C58826
Edition	00325-96020-68253-AAOEM
Version	64-bit operating system, x64-based processor
	Windows 10 Home 22H2
MATLAB®	'9.9.0.1718557 (R2020b) Update 6'. Student License 40968122
ThingSpeak™	1 Unidad (10 channels) 60 mensajes/1min. c/1seg.
	License: 41284718

3. The Image Processing Toolbox de MATLAB®

The Image Processing Toolbox ofrece un amplio conjunto de funciones morfológicas que se usaron para implementar los algoritmos para segmentar las imágenes de la espuma. Se segmentaron las regiones de interés de la espuma para medir las características de la espuma que se muestran en la Tabla 7.

Tabla 7

Características físicas, estáticas y dinámicas de la espuma

Tipos de características	Variables dependientes
1. Características físicas	• Tamaño de la burbuja • Color de la espuma
2. Características estáticas	• Textura: Energía
3. Características dinámicas	• Estabilidad de la espuma • Velocidad de la burbuja

4. Algoritmos de segmentación

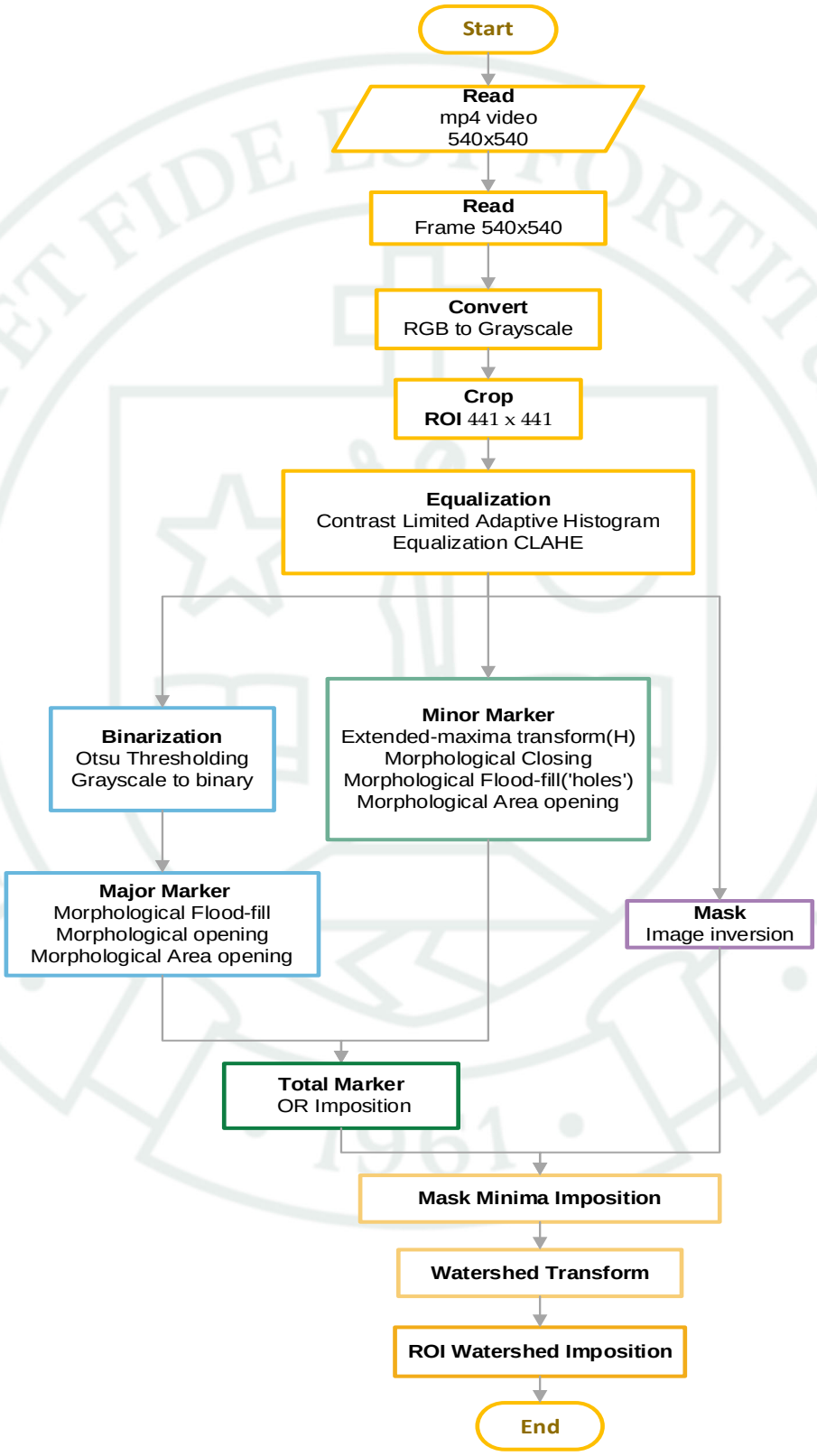
Se implementó los algoritmos de segmentación para obtener las características de la espuma. A continuación, se detallarán los algoritmos.

4.1 Método de segmentación propuesto para medir el área de la burbuja

En las celdas de flotación de cobre o de oro, se añaden reactivos para volverlos hidrofóbicos para que se adhieran a las burbujas para recuperar el producto valioso. Sin embargo, la metalúrgica del hierro es inversa porque la separación magnética suele ir primero para concentrar el hierro y luego viene la flotación donde se aplican otros reactivos para que las impurezas tales como la sílice y el azufre se vuelvan hidrofóbicas con el fin de que se adhieran a las burbujas, pero el hierro se va para el fondo como relave purificado o producto final. Para ambos tipos de flotación se puede medir el tamaño de las burbujas. Luego, para estudios posteriores se puede controlar la dosis de reactivos en función al tamaño de las burbujas. En la Figura 14 se muestra el diagrama de flujo del método propuesto basado en la segmentación *watershed* controlada por marcadores (*marker-controlled watershed*) para medir el área de las burbujas.

Figura 14

Diagrama de flujo del método propuesto para medir el área de las burbujas



Nota. Se muestra el diagrama de flujo del método propuesto basado en la segmentación watershed controlada por marcadores (*marker-controlled watershed*) para medir el área de las burbujas

Para encontrar el área de la burbuja se adoptaron algunas funciones morfológicas y el algoritmo *watershed* porque es más robusto y efectivo que el algoritmo de detección de borde (*edge detection*) para realizar esta tarea (Sadr-Kazemi y Cilliers, 1998). Como se muestra en la Tabla 8 se obtuvieron las propiedades del video 7s_CLASE_A.mp4.

Tabla 8

Propiedades del video con sus valores

Propiedades del video	Valores
Número de frames	228
Duración	7.6430
Ancho	540
Alto	540
Frecuencia de frames	30
Bits por píxel	24
Formato de video	'RGB24'

El procesamiento de imágenes consiste en encontrar: 1) la región de interés ROI, 2) la ecualización del contraste del ROI, 3) la binarización, 4) los marcadores para las burbujas grandes, 5) los marcadores para las burbujas pequeñas, 6) la superposición de ambos marcadores para obtener un marcador total, 7) la máscara de la imagen ecualizada, 8) los mínimos regionales interpuestos en la máscara y 9) la transformada *watershed* superpuesta en el ROI. Todos estos pasos se lograron aplicando técnicas de

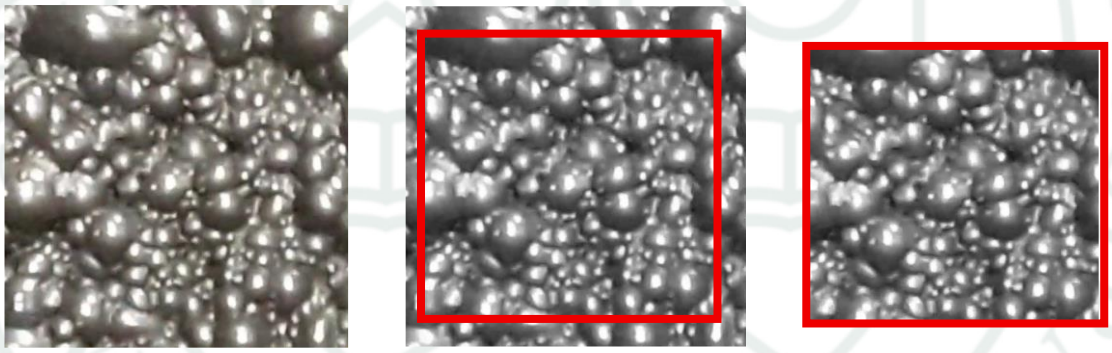
segmentación como morfologías de dilatación, erosión, apertura, cierre y otros algoritmos tales como la transformada de máximos extendidos, la imposición mínima y la transformada de *watershed*. La integración de estos últimos algoritmos permitió implementar el algoritmo *marker-controlled watershed*.

4.1.1 Región de interés (ROI)

Se escogió el frame 35 para realizar las pruebas de segmentación. Se convirtió la imagen a color, Figura 15 a), a escala de grises, Figura 15 b). Luego, se recortó esta imagen de 540 × 540 a 441 × 441 píxeles para obtener la región de interés (ROI) y para optimizar el procesamiento computacional, como se muestra en la Figura 15 c).

Figura 15

a) Espuma a color, b) Espuma en escala de grises, c) ROI de 441 × 441 píxeles



a) b) c)

4.1.2 Ecualización del ROI

Para mejorar el contraste de la imagen del ROI, Figura 16 a), se aplicó el algoritmo de ecualización adaptativa de histograma limitada por el contraste (CLAHE). Esta sería la imagen ecualizada, como se muestra en la Figura 16 b).

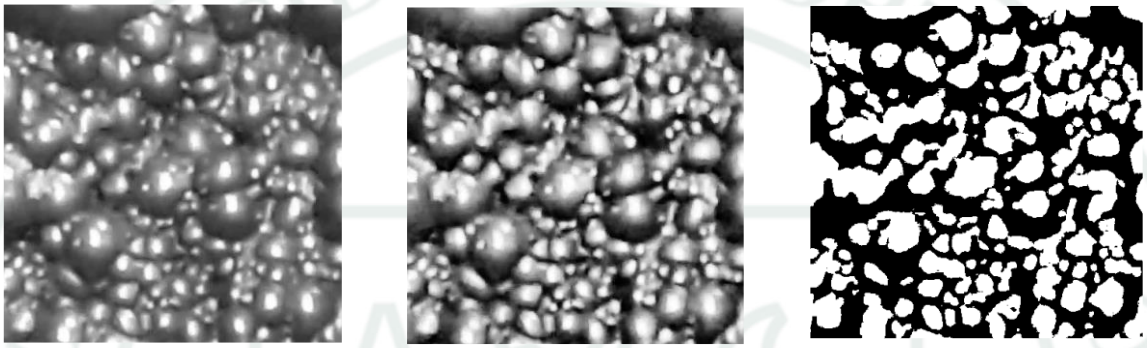
4.1.3 Binarización

Empleando el método de *Otsu*, se obtuvo el umbral de la imagen global igual a 0.5098. Luego se transformó a una imagen binaria sustituyendo todos los píxeles de la

imagen con una luminancia superior al umbral global por el valor 1 (blanco) y sustituyendo todos los demás píxeles por el valor 0 (negro), como se muestra en la Figura 16 c).

Figura 16

Binarización: a) ROI, b) Imagen ecualizada, c) Imagen binaria



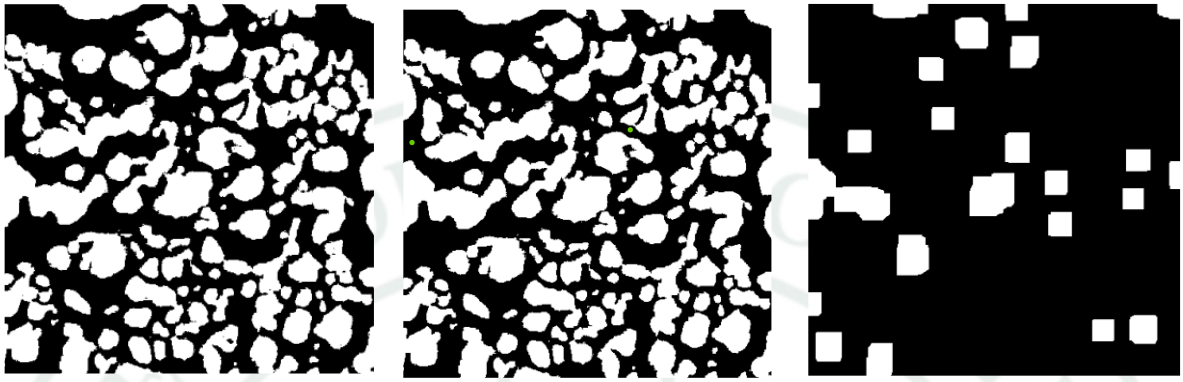
a) b) c)

4.1.4 Marcador mayor

El marcador mayor marca las burbujas más grandes. Para obtener el marcador mayor de la imagen binaria, Figura 17 a), se tuvo que aplicar operaciones morfológicas como la dilatación, el *flood-fill* para rellenar los agujeros, como se muestra en la Figura 17 b). Luego se aplicó una apertura morfológica que consiste en una erosión (eliminación de pequeños detalles) seguida de una dilatación. Finalmente, se aplicó una apertura para eliminar todos los componentes que tienen menos de 10 píxeles. Este fue el marcador mayor. como se muestra en la Figura 17 c).

Figura 17

a) Imagen binaria, b) Algoritmo flood-Fill, c) Marcador mayor



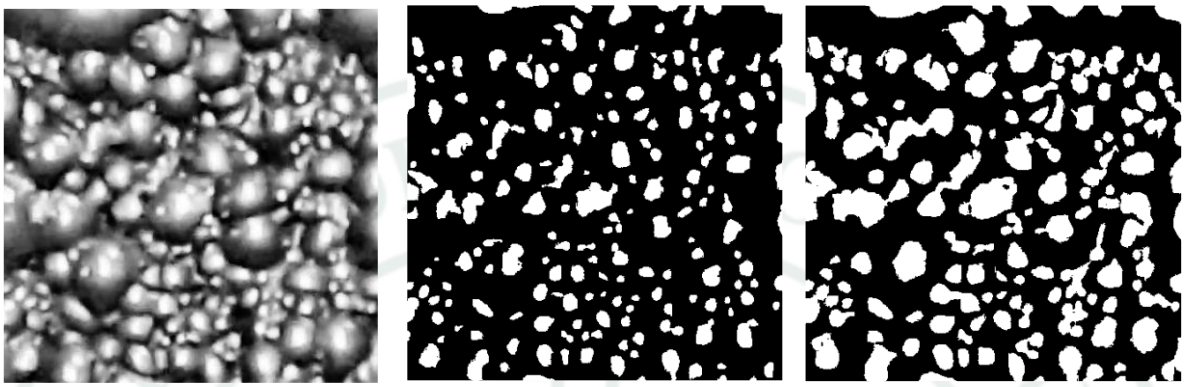
a) b) c)

4.1.5 Marcador menor

Se le llama marcador menor porque marca las burbujas más pequeñas. Para obtener la máxima extendida de la imagen ecualizada, Figura 18 a), se aplicó la transformada de máximos extendidos, Figura 18 b), que son los máximos regionales de la transformada H-máximos (escalar no negativo). Estos máximos regionales son los componentes conectados de píxeles con un valor de intensidad constante (el reflejo de la burbuja) y cuyos píxeles de límites externos tienen un valor inferior (los vecinos del reflejo de la burbuja). Cuando H es 50, el reflejo de la burbuja es más pequeño y cuando H es 90, es más grande como se muestra en la Figura 18 c).

Figura 18

a) Ecualización, b) Máxima extendida, H=50, c) Máxima extendida, H=90

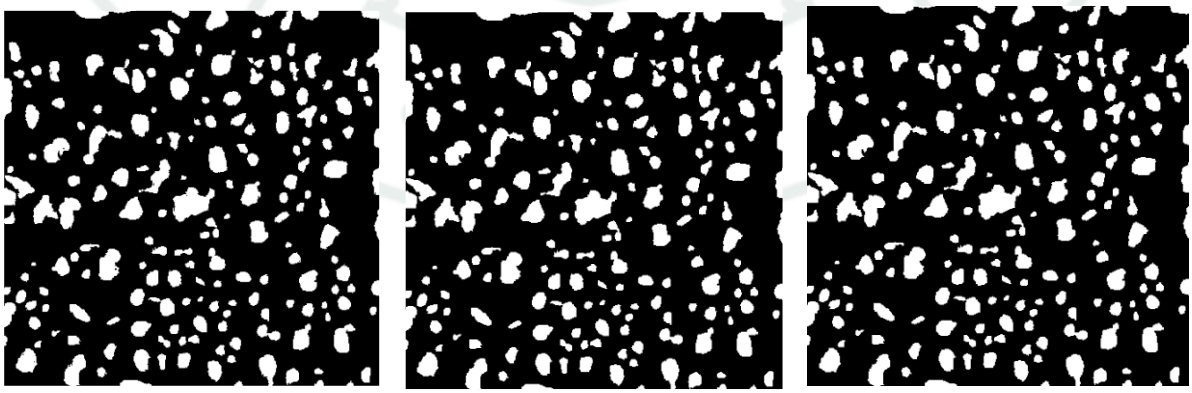


a) b) c)

Como se ve en la Figura 19 a), a la imagen con la máxima extendida con H=50 se le aplicó la operación de cierre. El cierre consiste en una dilatación seguida de una erosión. Luego, según la Figura 19 b), se le aplicó el algoritmo *flood-fill* para rellenar los agujeros, y finalmente, se le aplicó una apertura de área para eliminar todos los componentes que tienen menos de 5 píxeles, y este fue el resultado del marcador menor, como se muestra en la Figura 19 c).

Figura 19

a) Máxima extendida, H=50, b) Flood-Fill, c) Marcador Menor



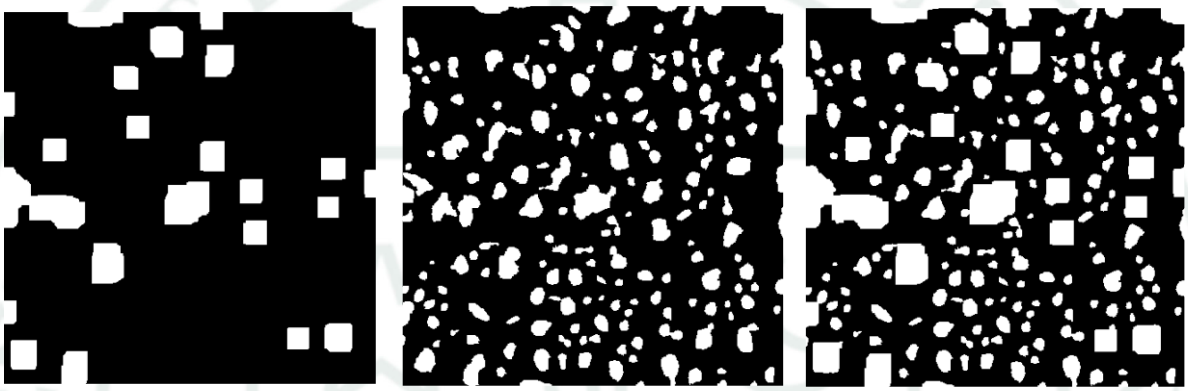
a) b) c)

4.1.6 Marcador total

Se aplicó la superposición del marcador mayor, Figura 20 a), sobre el marcador menor, Figura 20 b). La información de interés de ambas imágenes fue el marcador total, como se muestra en la Figura 20 c).

Figura 20

a) Marcador mayor, b) Marcador menor, c) Marcador total



a) b) c)

4.1.7 Máscara

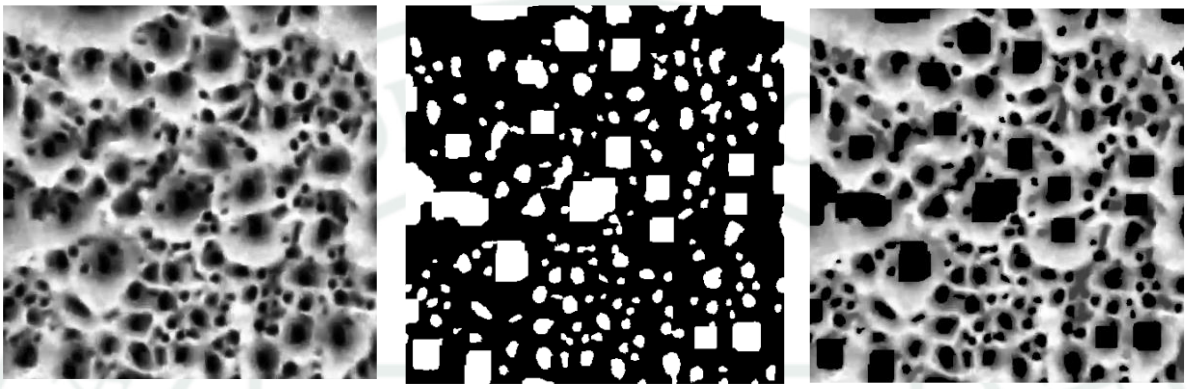
Se aplicó un complemento a la imagen ecualizada para obtener la máscara como se muestra en la Figura 21 a).

4.1.8 Imposición mínima en la máscara

Se aplicó el algoritmo de imposición mínima. Este algoritmo modifica la imagen de la máscara como se muestra en la Figura 21 a), utilizando la reconstrucción morfológica de modo que solo tenga mínimos regionales donde la imagen del marcador sea distinta de cero (áreas blancas del marcador) como se muestra en la Figura 21 b). Aquí se interpuso el mínimo regional del marcador total con la máscara, donde las áreas blancas del marcador total se convierten en áreas negras y luego se interpuso sobre la máscara como se muestra en la Figura 21 c).

Figura 21

a) La máscara de la imagen ecualizada, b) Marcador total, c) Imposición Mínima del marcador total sobre la máscara



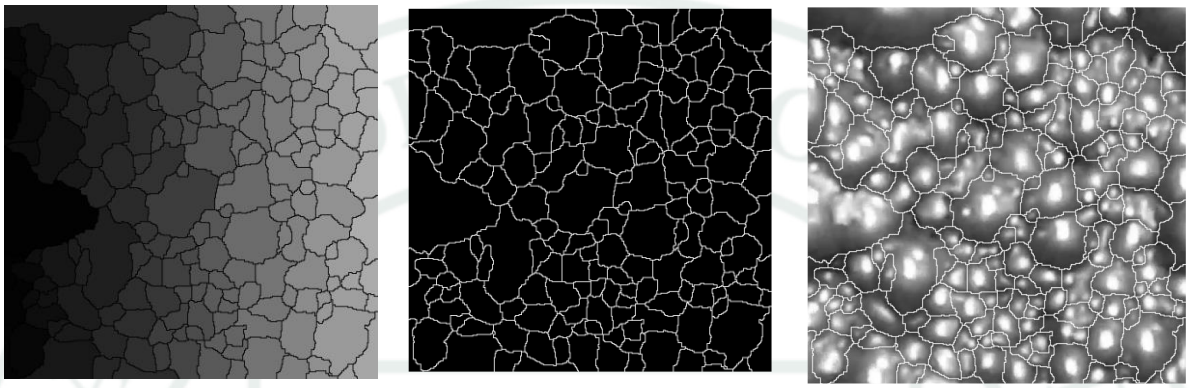
a) b) c)

4.1.9 Transformada watershed superpuesta en la ROI

Ahora se aplicó la transformada *watershed*. Este algoritmo encuentra las líneas de cresta de las cuencas donde los píxeles claros representan elevaciones altas y los píxeles oscuros representan elevaciones bajas (The MathWorks, Inc., 2025a). Este método que se implementó está basado en la segmentación *watershed* controlada por marcadores (*marker-controlled watershed*) para encontrar los perímetros de las burbujas, ya que es de interés para poder hallar el área de las burbujas. Primero obtuvimos la matriz que identifica las regiones de *watershed* de la imagen con la interposición mínima de regiones, como se muestra en la Figura 22 a). Se construyó una imagen binaria *watershed* extrayendo las líneas *watershed* con una comparación lógica de igualdad a 0. Los píxeles que tenían valor 0, pasaron a 1 como se muestra en la Figura 22 b). Finalmente, se interpuso las líneas blancas *watershed* que tienen valor 1 igualando con un valor de 255 para ver las líneas blancas *watershed* sobre la imagen de la región de interés ROI, como se muestra en la Figura 22 c).

Figura 22

a) Watershed en función de la interposición mínima, b) Imagen binaria watershed, c) watershed superpuesta en la ROI

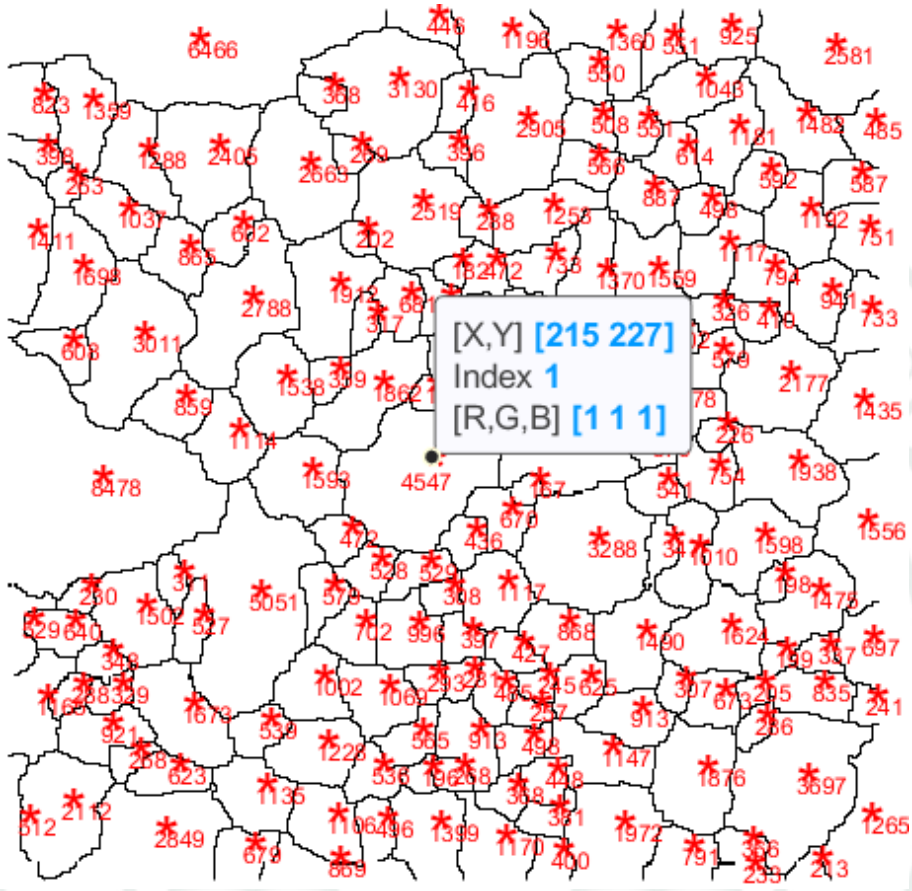


a) b) c)

Luego, se invirtieron las líneas de *watershed* de blanco a negro haciendo un *not* lógico para poder obtener las propiedades tales como el área y el centroide de las burbujas. Se superpusieron los centroides con sus áreas. Por ejemplo, vemos que en el centroide (x = 215, y = 227), su área es de 4547 píxeles como se muestra en la Figura 23.

Figura 23

Centroides (*) y áreas (píxeles) de las burbujas



4.1.10 Escalamiento del área en cm²

El área es el número de píxeles en la región encerrada. Por lo tanto, el área ya está en píxeles² y si se desea medir el área en cm², se tiene que aplicar un factor de escalamiento (*scale*), que es la relación que hay entre la distancia experimental en cm y la distancia en píxeles como se muestra en la Ecuación 1.

$$S = \frac{d}{x}$$

$$S = \frac{d}{xf\ - xi} \qquad (1)$$

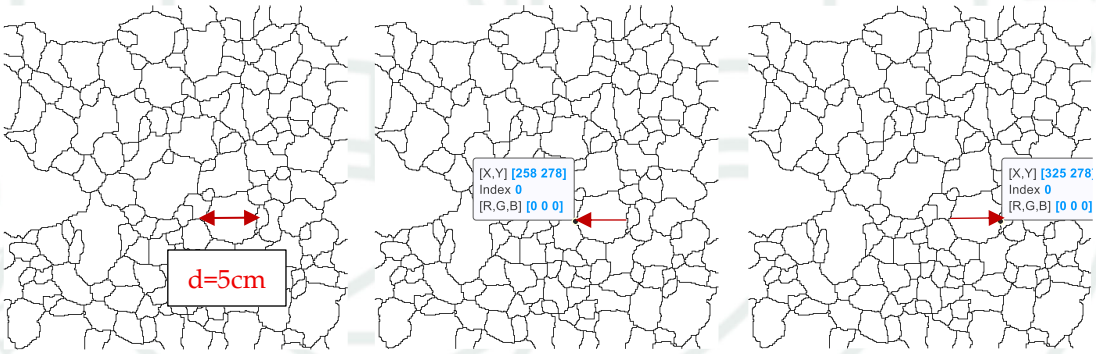
Donde:

- S = escalamiento del área (en centímetros por píxeles, cm/px);
- d = distancia experimental (en centímetros, cm), Figura 24 a);
- xi = posición inicial de la burbuja (en píxeles, px), Figura 24 b);
- xf = posición final de la burbuja (en píxeles, px), Figura 24 c);
- x = distancia entre xf y xi (en píxeles, px).

Cuando se tomaron las muestras, no se pudo poner la regla en la celda de flotación. Sin embargo, se usó una distancia referencial como se muestra en la Figura 24 a).

Figura 24

a) Distancia en cm, b) Izquierda de la burbuja, c) Derecha de la burbuja



a) b) c)

Si aplicó la Ecuación 1, vemos que, para una distancia de 5 cm, hay 67 píxeles porque se resta $325 - 258$ como se muestra en la Figura 24 b) y Figura 24 c). La razón fue 1 cm /13.4 px.

$$S = \frac{5}{325 - 258}$$

$$S = \frac{5}{67} = \frac{1}{13.4}$$

Si se moviera la posición de la cámara a una altura más alta o baja, solo tendríamos que calibrar usando la Ecuación 1.

Por otro lado, el área en cm^2 sería el producto del área en píxeles por el factor de escalamiento elevado al cuadrado como se muestra en la Ecuación 2.

$$A = a \times S^2$$

$$A = a \times \left(\frac{d}{x}\right)^2 \quad (2)$$

Donde:

- A = área de la burbuja (en centímetros cuadrados, cm^2);
- a = área de la burbuja (en píxeles cuadrados, px^2);
- x = distancia entre xf y xi (en píxeles, px);
- d = distancia experimental (en centímetros, cm);
- S = escalamiento del área (en centímetros por píxeles, cm/px).

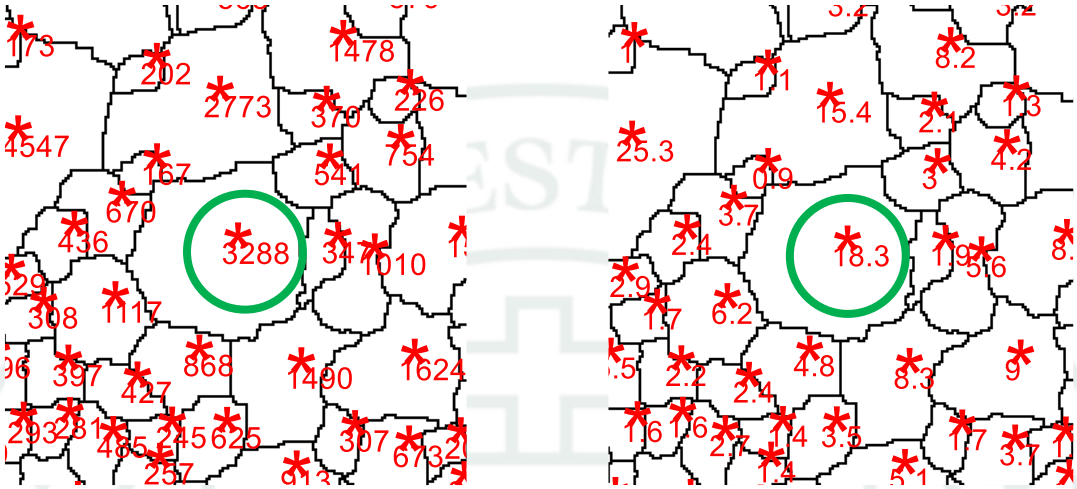
En la Figura 25 a), el área de la burbuja con círculo verde fue 3288 píxeles. Se aplicó la Ecuación 2, vemos que el área fue 18.3cm^2. Se validó este resultado en la Figura 25 b).

$$Area = 3288\ (pixeles^2) \times \left(\frac{5\ (cm)}{67\ (pixeles)}\right)^2$$

$$Area = 18.3\ cm^2$$

Figura 25

a) Áreas en píxeles, b) Áreas en cm²



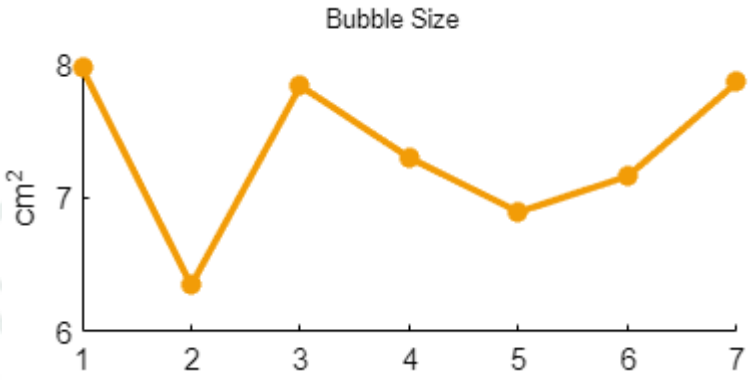
a) b)

4.1.11 Tendencias para la medición del área

Luego se obtuvo el promedio de las burbujas de todo el frame. Es importante recalcar que el video es de 30 fps. Se consideraron los dos primeros frames para realizar la primera medida del área. No se consideró el segundo y el tercer frame para hacer la segunda medida porque implicaría más procesamiento, retardos y si se hubiera hecho no hubiera sido muy significante porque al final el resultado era el promedio de los promedios. Por ende, para la segunda medida se consideraron el tercer y cuarto frame, para la tercera medida el quinto y sexto frame y así sucesivamente. Finalmente, se promedió quince medidas para poder tener una medida promedio del área para un segundo del video. En la Figura 26, se mostraron siete segundos del video 7s_CLASE_A.mp4 con sus respectivas medidas del área en cm² donde el intervalo entre muestras medibles fue de 1 segundo de video.

Figura 26

Medición del área de las burbujas durante 7 segundos



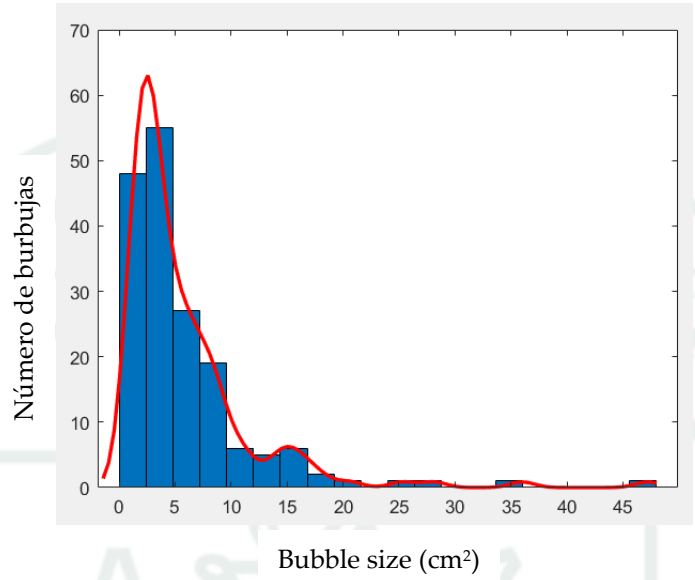
La plataforma IoT que se escogió es ThingSpeak™. Esta plataforma es gratuita para intervalos de 15 segundos. Sin embargo, se tuvo que pagar una suscripción de $45.00 para poder enviar datos cada segundo.

4.1.12 Distribución del tamaño de las burbujas (BSD)

Se hizo medición de la distribución del tamaño de las burbujas (BSD) basándose en un histograma donde en el eje *x* tenemos el tamaño del área de las burbujas en cm^2 y en el eje *y* el número de burbujas o frecuencia que hay en función al área en intervalos (bins) de 20. Se usó una línea roja de tendencia de densidad Kernel para poder visualizar mejor la distribución. Se pudo interpretar la Figura 27 donde vemos que 55 burbujas midieron en el rango de 2.5 cm^2 a 5 cm^2, y 28 burbujas midieron en el rango de 5 a 7.5 cm^2, etc. Se observó que hay más burbujas pequeñas que grandes. Se pudo observar que hay 2 burbujas de aproximadamente 35 cm^2, etc. En el capítulo V se muestra más detalles sobre los resultados.

Figura 27

Distribución del tamaño de las burbujas (BSD) para el frame 35



Para medir la velocidad de la espuma se necesitaba medir el desplazamiento de los centroides en el siguiente frame. Sin embargo, algunas burbujas colapsaban. Por ende, primero se implementó un algoritmo para determinar la estabilidad de las burbujas y luego se implementó el algoritmo para la velocidad usando las burbujas estables como referencia.

4.2 Algoritmo para medir la estabilidad de la espuma

La caracterización de la inestabilidad de la espuma se basa en la tasa de colapso de la burbuja en la espuma.

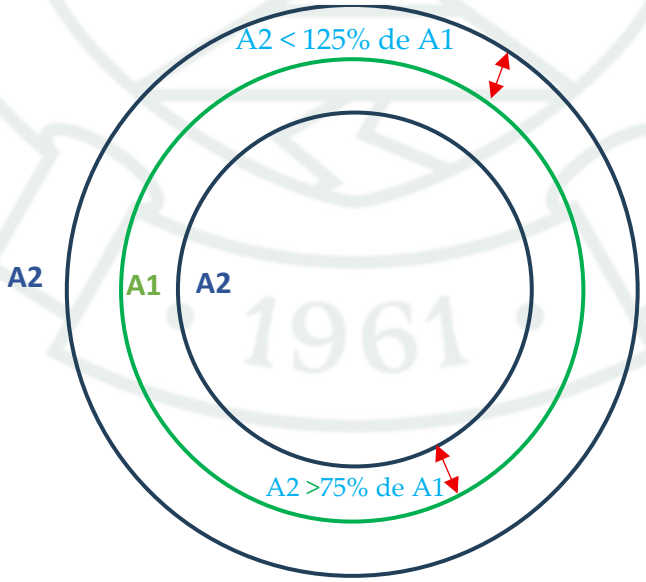
4.2.1 Criterio para detectar la burbuja estable

En esta investigación se encontró otro método para medir la estabilidad basado en los centroides y el tamaño de las burbujas. Primero, entendimos que las burbujas más grandes son las que llegan a colapsar dejando burbujas más pequeñas en su reemplazo, lo cual hace que esa zona sea más oscura. Sin embargo, no se consideraron las zonas oscuras porque cuando se sacó el promedio en escala de grises del primer

frame y se comparó con el segundo frame, se notó que no guardaban una relación. A veces el primer frame era más oscuro que el segundo o viceversa. Por lo tanto, se obtuvieron los centroides del primer y del segundo frame de las 10 burbujas más grandes para detectar cuantas colapsan. En la Figura 28, vemos dos burbujas donde A1 representa a una burbuja verde del primer frame con su área y sus coordenadas *x*, *y*. De la misma manera, A2 representa la burbuja azul del segundo frame con su área y sus coordenadas. El tiempo entre dos frames es 1/30 = 0.033 s. La burbuja verde A1 se volverá en la burbuja azul A2 en un lapso de 0.033 s si es estable y puede que aumente o disminuya aproximadamente en un ±25 %. Si la burbuja A1 no se convierte en una burbuja A2, es porque ha colapsado y, por lo tanto, A1 no es estable. A1 permanecerá estable en el siguiente frame para volverse A2 si el área de A2 es mayor al 75 % de A1 y menor al 125 % de A1. Además, la posición *x* de A2 debe ser aproximadamente mayor al 75 % de la posición *x* de A1 y menor al 125 % de la posición *x* de A1.

Figura 28

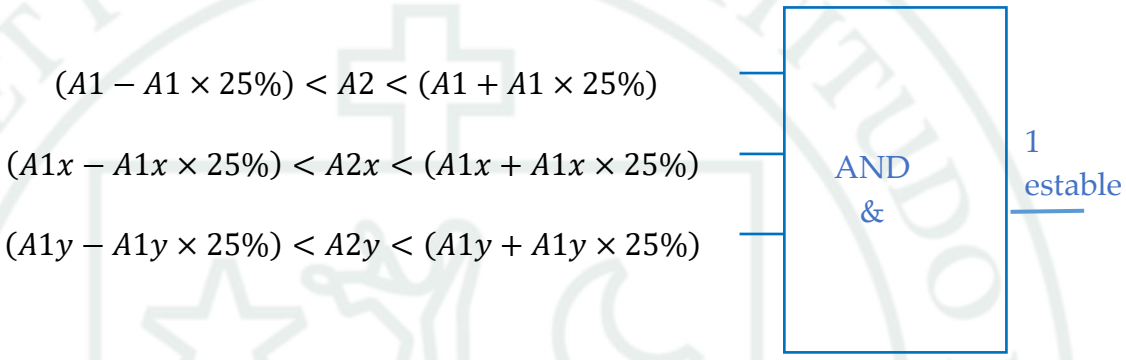
Criterio para detectar la burbuja estable en el 2do frame



Y también debe cumplir que la posición y de A2 debe ser aproximadamente mayor al 75 % de la posición y de A1 y menor al 125 % de la posición y de A1, como se muestra en la Figura 29, donde vemos un *interlock* de tipo *and* lógico. Si una de ellas no cumple, entonces la burbuja A1 no es estable.

Figura 29

Interlock para la estabilidad de la burbuja A1

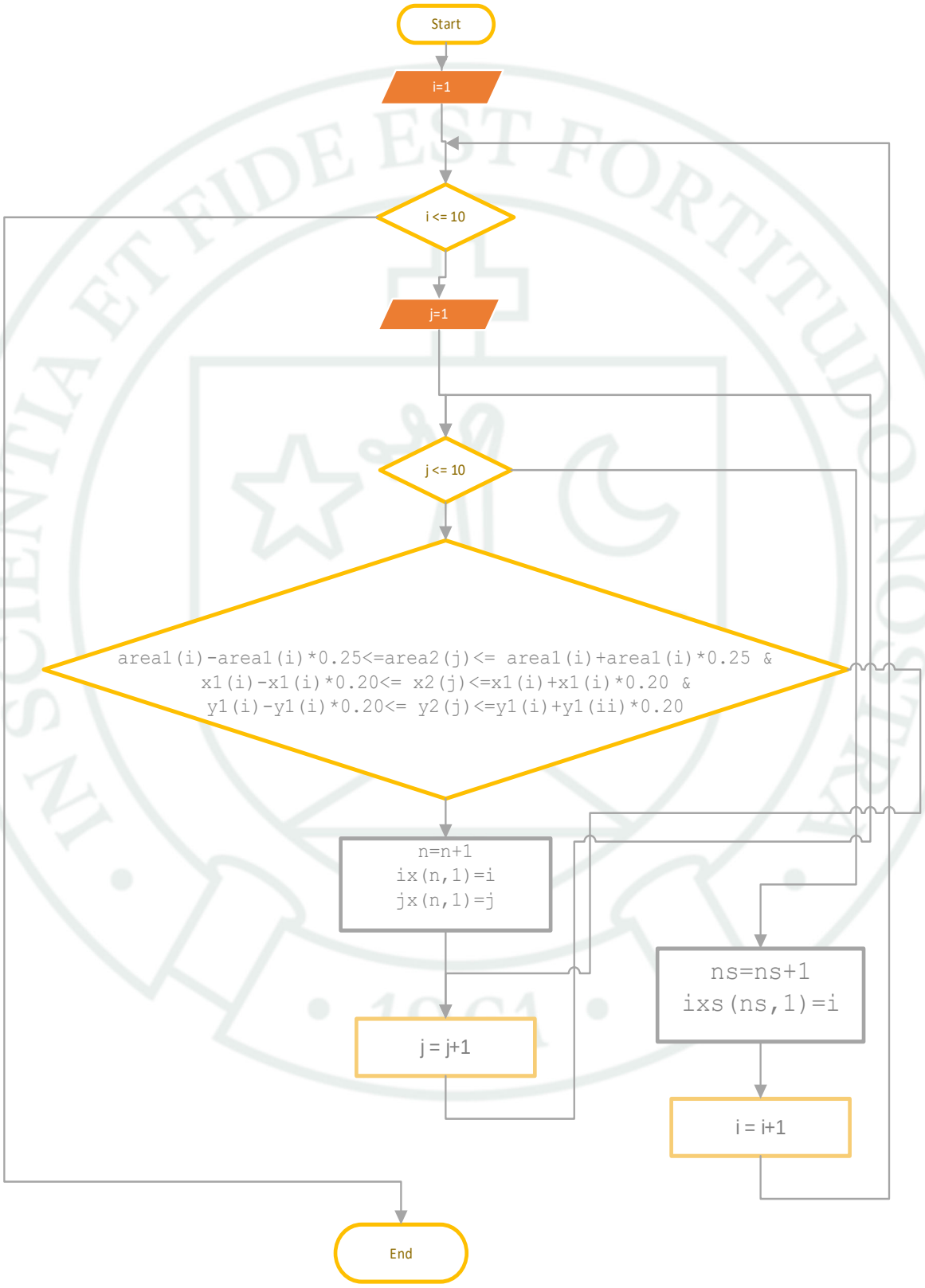


Si la respuesta del *interlock and* es 0, entonces esa burbuja A1 es inestable. Para poder ver la estabilidad de las diez burbujas más grandes del frame 1, se tuvo que aplicar dos estructuras *for* y aplicar el *interlock and* para identificar los índices de las burbujas estables e inestables. A todas las burbujas del primer y segundo frame, se les asignó un índice para que sean identificadas.

La Figura 30 nos muestra el diagrama de flujo para identificar los índices de las burbujas estables y los índices de las burbujas inestables del frame 1, donde i es el índice de la burbuja del frame 1 y j es el índice de la burbuja del frame 2. Esta burbuja del frame 2 representa la burbuja del frame 1 después de un tiempo de 0.033 s, ya que es estable. Si el *interlock and* es un 1 lógico, entonces es estable y me devuelve el índice i en ix de la burbuja estable del frame 1 y el índice j en jx de la burbuja del frame 2. Si el *interlock and* es un 0 lógico, me devuelve el índice i en ixs de la burbuja inestable del frame 1.

Figura 30

Identificación de los índices de las burbujas estables e inestables, frame 1



El algoritmo pudo detectar las 10 burbujas más grandes que tienen el mismo tamaño y posición dentro de un rango de ± 25%. Luego de aplicar el algoritmo al video 7s_CLASE_A.mp4, se mostraron los resultados en la Tabla 9, donde el primer frame es 35 y el segundo frame es 36.

Tabla 9

Índices de las burbujas estables del 1er y 2do frame

Frame 35				*Frame 36*			
Índice i	***Área cm²***	***x***	***y***	***Índice j***	***Área cm²***	***x***	***y***
1	47	43	242	1	40	95	22
2	36	92	20	2	35	91	161
3	28	123	300	3	28	218	240
4	25	211	233	4	27	122	309
5	21	400	393	5	25	68	248
6	18	294	273	6	24	19	239
7	17	193	40	7	20	403	399
8	17	64	169	8	19	297	277
9	16	258	60	9	19	196	44
10	6	75	420	10	17	385	241

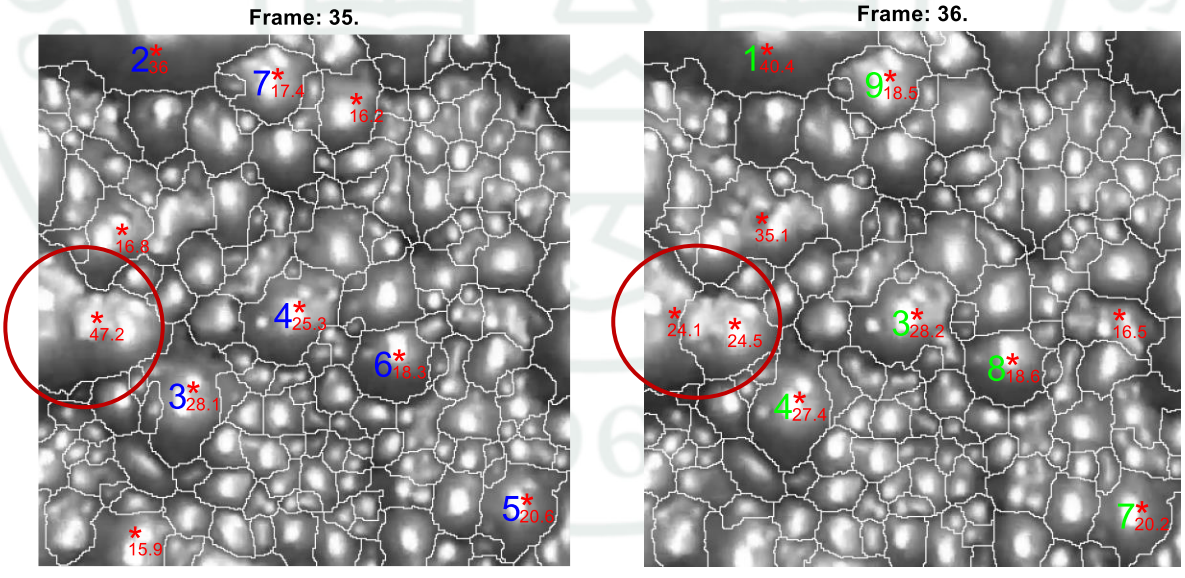
Vemos que cuando el índice *i* es 2, su área es 36 y su posición es (92,20), y esta burbuja se ve en el frame 2 cuando su índice *j* es 1, su área es 45, y su posición es (95,25). Esto indica que esta burbuja apareció en el frame 2 y se le puso color rojo para identificarla como estable. Vemos como los índices *i* y *j* están mapeados por colores porque son estables y tienen relación de área y posición en un ±25% entre ellos. Los índices *i*, 1, 8, 9 y 10 muestran a las burbujas inestables, ya que no se encontraron esas

burbujas con esas mismas características de tamaño y posición en el frame 2 porque lo más probable es que han colapsado o se han unido con otras o se han dividido en 2 o más. El algoritmo solo identifica las burbujas que han colapsado, más no a las que se han unido con otras burbujas haciéndose más grandes ni a las que se han dividido por sobresegmentación, lo cual sería un tema para futuras investigaciones.

En la Figura 31 a), vemos que los índices del frame 35 están en azul, y en la Figura 31 b), vemos los índices detectados del frame 36 que están en verde. Según la Tabla 9, y la Figura 31, observamos que en el índice 2 del frame 35 tenemos una burbuja con área 36, y después esta burbuja aumenta su área a 40 pero se encuentra en el índice 1 del frame 36, lo cual nos indica que es estable. El índice 3 del frame 35 muestra un área de 28 y después lo vemos en el índice 4 del frame 36, pero esta vez su área ha disminuido a 27 y también es estable.

Figura 31

Burbujas estables mapeadas entre el, a) 1er frame, y el b) 2do frame



a) b)

En conclusión, vemos que los índices 2, 3, 4, 5, 6 y 7 del frame 35 en la Figura 31 a) están mapeados respectivamente con los índices 1,4,3,7,8 y 9 del frame 36, como se muestra en la Figura 31 b). Vemos que 6 de 10 burbujas se detectaron en los frames 35 y 36; por lo tanto, se determinó una estabilidad del 60 %. En la Ecuación 3, podemos expresar la estabilidad en porcentaje.

$$E = \frac{n}{N} \times 100 \tag{3}$$

Donde:

- E = estabilidad de las burbujas (en porcentaje, %);
- n = número de burbujas mapeadas en el 1er y 2do frame;
- N = número de las burbujas más grandes (N = 10).

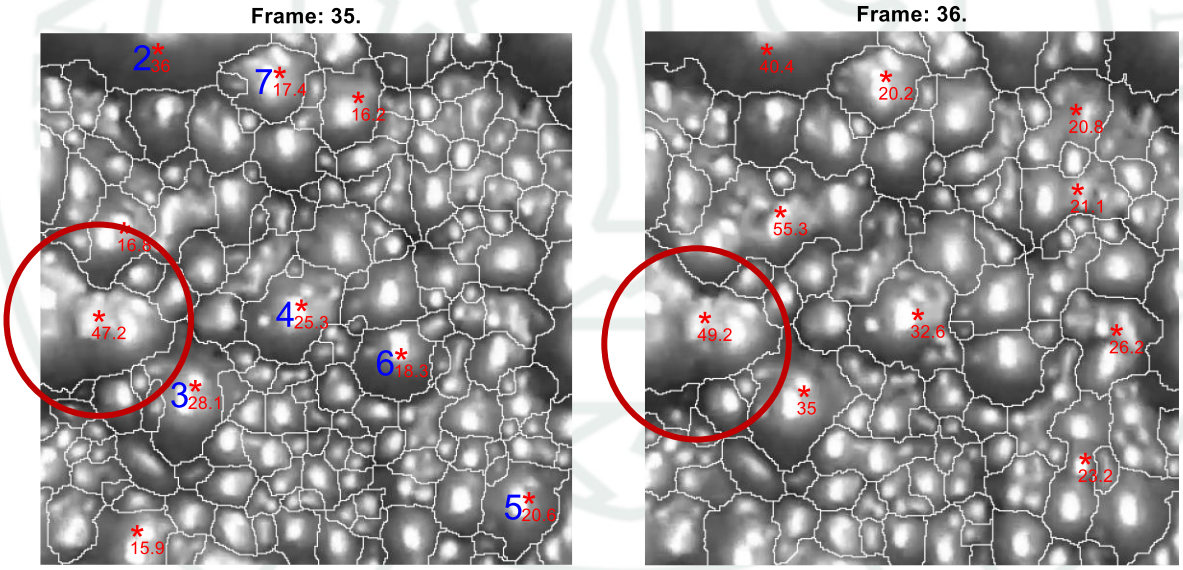
4.2.2 Sobresegmentación y subsegmentación en la medición de la estabilidad

Hubo algunos inconvenientes en el análisis de la estabilidad por la complejidad de la segmentación, la dinámica de la espuma y el reflejo de la luz sobre las burbujas. Cuando se dividen las burbujas, ocurre el fenómeno de fragmentación de burbujas y cuando se unen, ocurre el fenómeno de coalescencia. Estos fenómenos pueden ocurrir físicamente por la dinámica de la espuma o pueden ocurrir artificialmente debido a la sobresegmentación y subsegmentación, respectivamente. Se vio en la Figura 31 a), que la burbuja del frame 35 con área 16.8 se unió con otras burbujas en el siguiente frame 36, como se ve en la Figura 31 b), haciendo que su área sea 35, lo cual lo hace más grande. Este problema parece ser debido a los cambios de luz en esa área, por lo que se considera un fenómeno de coalescencia artificial de burbujas generado por la luz. La burbuja con área igual a 47 del frame 35, Figura 31 a), se ve que en realidad son 2 burbujas, pero el algoritmo las ve como una sola burbuja por la luz. Luego, en el frame

36, esta burbuja se fragmentó correctamente en 2 burbujas con áreas de 24 y 24.5 por la claridad de la línea de cresta, haciendo que se logre una buena segmentación, como se muestra en la Figura 31 b), con círculos rojos, debido al parámetro H = 50 de la transformada máxima extendida que se usó para crear el marcador menor. Esa fragmentación es física y no por la sobresegmentación. Sin embargo, si cambiamos el H=95 para el frame 36 que es el segundo frame, vemos que el problema sería que varias burbujas se empiezan a unir entre ellas debido a la subsegmentación o coalescencia artificial, como se muestra en la Figura 32 b). Cuando H disminuye a 10, ocurre una sobresegmentación, es decir se empiezan a fragmentar artificialmente las burbujas.

Figura 32

a) Frame 35 con H=50 y b) Frame 36 con área unida, H=95



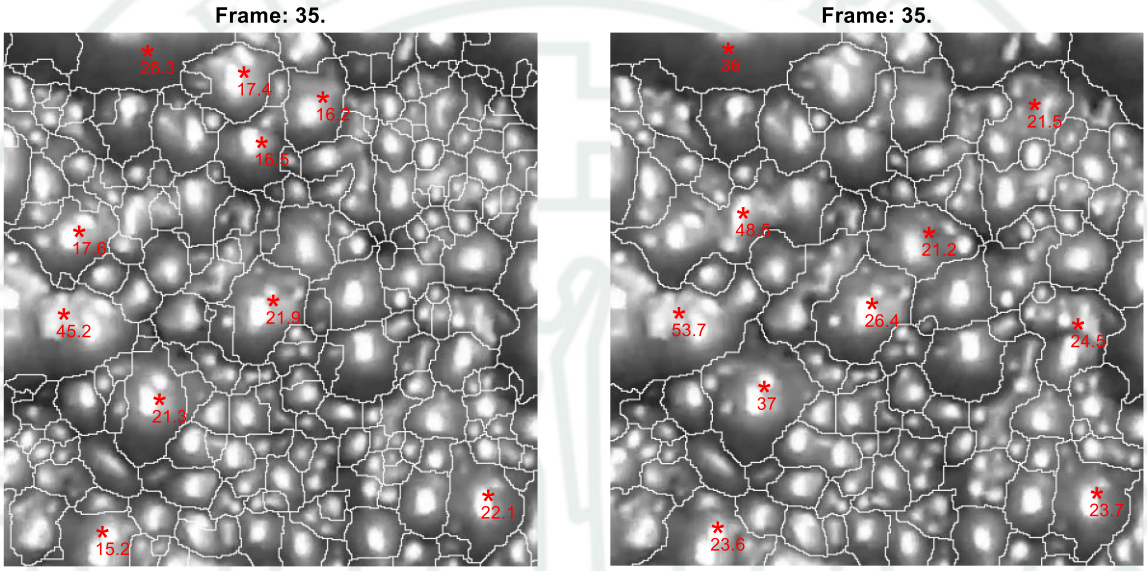
a) b)

Analizando el frame 35, vemos que la sobresegmentación ocurre cuando usamos un H muy bajo, por ejemplo, H=5, donde se detectan las burbujas más pequeñas, y a veces empieza a fragmentar las burbujas artificialmente, lo cual hace que el número de burbujas aumente, como se observa en la Figura 33 a). La

subsegmentación ocurre cuando usamos un H muy alto, ejemplo H=95, haciendo que se pierdan algunas fronteras entre burbujas adyacentes y ocasionando un fenómeno de coalescencia artificial como se muestra en la Figura 33 b).

Figura 33

a) Sobresegmentación con H=5 y b) Subsegmentación con H=95



a) b)

Estos problemas podrían mitigarse mediante estrategias de segmentación adaptativa. Por ejemplo, el parámetro H podría variar en función del número y tamaño de las burbujas. Asimismo, podrían emplearse técnicas de segmentación semántica basadas en machine learning o redes neuronales, además de mejorar las condiciones de adquisición de imagen mediante una iluminación más controlada, cámaras de mayor calidad y una mayor capacidad de procesamiento. Estos aspectos podrían constituir líneas de investigación futuras. No obstante, las limitaciones identificadas no impidieron la visualización y seguimiento de las características de la espuma, cumpliendo así con el objetivo principal de esta investigación de integrar dicha

información en un sistema SCADA y un Dashboard IoT. En el capítulo de resultados se detallarán las medidas de estabilidad, colapso y coalescencia.

En resumen, la estabilidad se determina a partir de los centroides y las áreas de cada burbuja identificada en el frame. Posteriormente, mediante el mapeo entre frames consecutivos, se establecen las relaciones entre burbujas, identificando aquellas que permanecen estables. Una vez definidas estas burbujas, su velocidad se calcula a partir del desplazamiento de sus centroides entre ambos frames, considerando un intervalo temporal de 0.033 s.

4.3 Algoritmo para medir la velocidad de la espuma

El *mass-pull* es la tasa de flujo de sólidos que se reporta al concentrado, la cual se ve afectada por cambios en la estructura, la estabilidad de la espuma, el flujo de aire y la profundidad de la espuma (Hadler *et al*., 2010).

Por lo tanto, si medimos la velocidad de la espuma, podemos determinar el *mass-pull*, ya que están relacionados. En función de la velocidad medida, los controles de procesos ajustan la profundidad de la espuma para lograr la velocidad y el *mass-pull* requerido. (Supomo *et al*., 2008)

Para medir la movilidad o velocidad de la espuma en cm/s, se puede calcular siguiendo el movimiento de las burbujas en frames sucesivos. Los algoritmos más frecuentemente usados para medir la velocidad son *block matching* y el *bubble tracking* (Forbes, 2007).

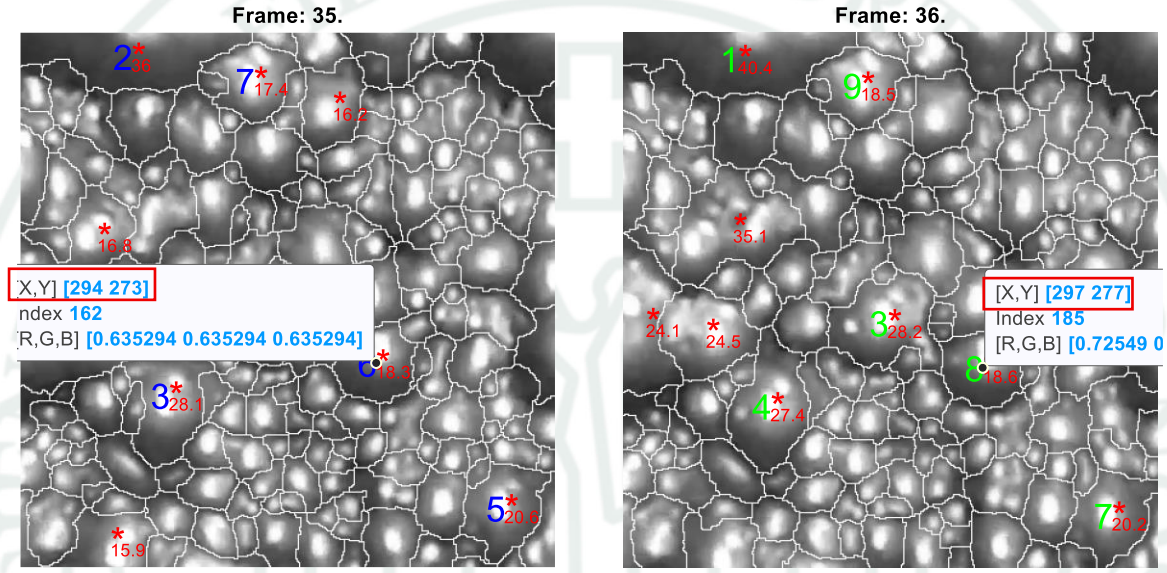
En esta tesis, para determinar la velocidad de las burbujas, primero se aplicó el algoritmo para detectar a las burbujas estables. Segundo, se tomó como referencia la posición de las burbujas mapeadas y estables del primer frame actual para luego evaluar el desplazamiento de sus centroides en el segundo frame consecutivo.

4.3.1 Vector desplazamiento

En la Figura 34, se muestran los centroides *(x, y)* de los índices 6 y 8 para determinar la velocidad de la burbuja con área de 18 cm^2.

Figura 34

a) Centroide del índice 6, frame 35 y b) Centroide del índice 8, frame 36

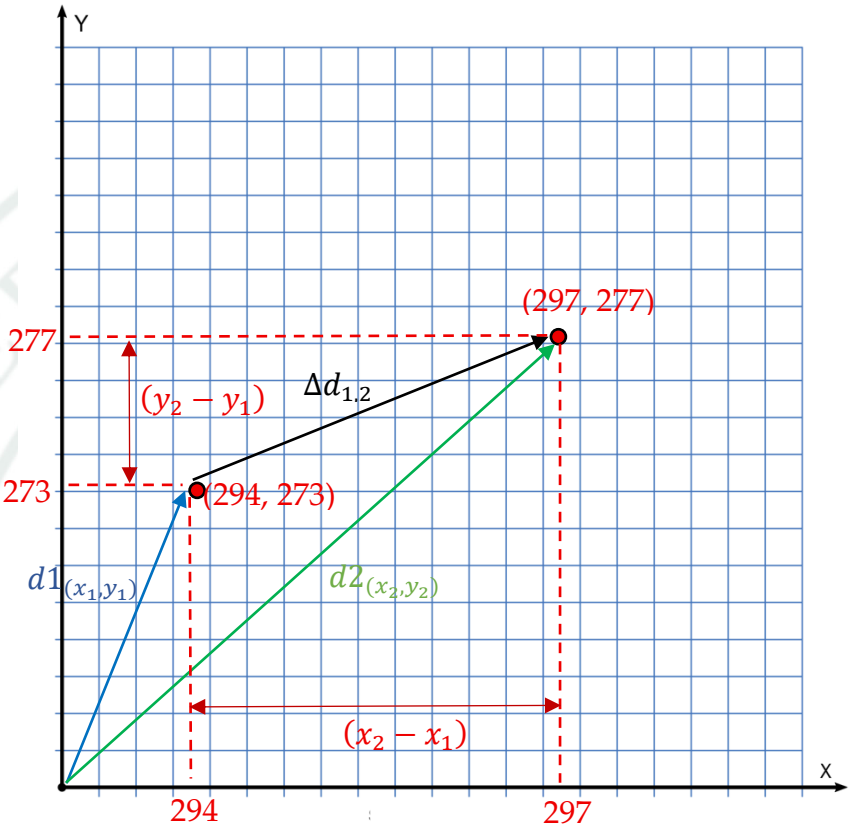


a) b)

Se analizaron los puntos de los centroides para determinar el vector desplazamiento $\Delta d_{1,2}$ como se muestra en la Figura 35.

Figura 35

Análisis para el vector desplazamiento $\Delta d_{1,2}$



El vector desplazamiento se expresa mediante la Ecuación 4.

$$\Delta d_{1,2} = d2_{(x_2,y_2)} - d1_{(x_1,y_1)}$$

$$\Delta d_{1,2} = (x_2 - x_1), (y_2 - y_1) \quad (4)$$

Donde:

- $\Delta d_{1,2}$ = vector desplazamiento (en píxeles, px);
- $d1_{(x_1,y_1)}$ = vector de posición del centroide con índice 6 del frame 35;
- $d2_{(x_2,y_2)}$ = vector posición del centroide con índice 8 del frame 36;
- x_1= posición x del 1er centroide, frame 35 (en píxeles, px);
- x_2 = posición x del 2do centroide, frame 36 (en píxeles, px);
- y_1= posición y del 1er centroide, frame 35 (en píxeles, px);

- y_2 = posición y del 2do centroide del frame 36 (en píxeles, px).

En la Tabla 10, se observa en azul el centroide *(x, y)*, el área del índice *i* 6 del frame 35 y, en verde, el centroide *(x, y)* y el área del índice *j* 8 del frame 36.

Tabla 10

Burbujas mapeadas y estables con sus índices, áreas y centroides

Frame 35				***Frame 36***			
Índice i	***Área cm²***	***x_1***	***y_1***	***Índice j***	***Área cm²***	***x_2***	***y_2***
2	36	92	20	1	40	95	22
3	28	123	300	4	27	122	309
4	25	211	233	3	28	218	240
5	21	400	393	7	20	403	399
6	18	294	273	8	19	297	277
7	17	193	40	9	19	196	44

Se aplicó la Ecuación 4, donde se restaron componentes *x* y los componentes *y* para obtener el vector de desplazamiento.

$$\Delta d_{1,2} = (x_2 - x_1), (y_2 - y_1)$$

$$\Delta d_{1,2} = (297 - 294), (277 - 273)$$

$$\Delta d_{1,2} = (3.422\,, 3.875)$$

4.3.2 Vector de velocidad media

Después de obtener el vector de desplazamiento $\Delta d_{1,2}$, se encontró el vector de velocidad media, como se muestra en la Ecuación 5.

$$\overrightarrow{Vm_{x,y}} = \frac{\Delta d_{1,2}}{\Delta t_{1,2}}$$

$$\overrightarrow{Vm_{x,y}} = \frac{(x_2 - x_1), (y_2 - y_1)}{(t_2 - t_1)} \quad (5)$$

Donde:

- $\overrightarrow{Vm_{x,y}}$ = vector de velocidad media (en píxeles por segundo, px/s);
- $\Delta d_{1,2}$ = vector de desplazamiento (en píxeles, px);
- t_1 = tiempo del 1er frame (en segundos, s);
- t_2 = tiempo del 2do frame (en segundos, s);
- $\Delta t_{1,2}$ = duración del tiempo entre frames (en segundos, s);
- x_1= posición x del 1er centroide, frame 35 (en píxeles, px);
- x_2 = posición x del 2do centroide, frame 36 (en píxeles, px);
- y_1= posición y del 1er centroide, frame 35 (en píxeles, px);
- y_2= posición y del 2do centroide, frame 36 (en píxeles, px).

Si consideramos t_1 igual a 0, entonces, t_2 es la razón de un frame sobre 30 frames por segundo, lo cual nos da 0.033 s.

$$t_1 = 0s$$

$$Frames\ por\ segundo\ (fps) = 30\frac{f}{s}$$

$$t_2 = \frac{1\,f}{30\frac{f}{s}} = 0.03333\ s$$

Ahora remplazamos t_2 y aplicamos la Ecuación 5, el vector de velocidad media sería:

$$\overrightarrow{Vm_{x,y}} = \frac{(x_2 - x_1), (y_2 - y_1)}{(0.03333\ s)}$$

$$\overrightarrow{Vm_{x,y}} = \left(\frac{(x_2 - x_1)}{0.03333\ s}, \frac{(y_2 - y_1)}{0.03333\ s}\right)$$

Ahora remplazamos $\Delta d_{1,2}$ y aplicamos la Ecuación 5, el vector de velocidad media sería:

$$\overrightarrow{Vm_{x,y}} = \left(\frac{3.422}{0.03333\ s}, \frac{3.8754}{0.03333\ s}\right)$$

$$\overrightarrow{Vm_{x,y}} = (102.660), (116.250)\ \frac{pixeles}{seg}$$

4.3.3 Módulo de velocidad media

El módulo de velocidad media se muestra en la Ecuación 6.

$$|\overline{Vm}| = \sqrt{(Vm_x)^2 + \left(Vm_y\right)^2} \quad (6)$$

Donde:

- $|\overline{Vm}|$ = módulo de velocidad media (en píxeles por segundo, px/s);
- Vm_x = componente x del vector de velocidad media (en píxeles por segundo, px/s);
- Vm_y = componente y del vector de velocidad media (en píxeles por segundo, px/s).

Aplicando la Ecuación 6, vemos que el módulo de velocidad media es igual a:

$$|\overline{Vm}| = \sqrt{(102.660)^2 + (116.250)^2}$$

$$|\overline{Vm}| = 155.090 \frac{pixeles}{s}$$

El módulo de velocidad media en cm/s se muestra en la Ecuación 7.

$$S = \frac{5\ (cm)}{67(pixeles)} = \frac{1\ (cm)}{13.4(pixeles)}$$

$$|\overline{Vm}| = \left(\sqrt{(Vm_x)^2 + \left(Vm_y\right)^2}\right) \times S \quad (7)$$

Donde:

- $|\overline{Vm}|$ = Módulo de velocidad media (en centímetros por segundo, cm/s);
- Vm_x = componente x del vector de velocidad media (en píxeles por segundo, px/s);
- Vm_y = componente y del vector de velocidad media (en píxeles por segundo, px/s);
- S = factor de escalamiento, Ecuación 1 (en centímetros por píxeles, cm/px**).**

Aplicando la Ecuación 7, el módulo de velocidad media es igual a 11.574 cm/s.

$$|\overline{Vm}| = 155.090\ \frac{pixeles}{s} \times\ S$$

$$|\overline{Vm}| = 155.090\ \frac{pixeles}{s} \times \frac{5}{67}\ \frac{cm}{pixeles}$$

$$|\overline{Vm}| = 11.574\ \frac{cm}{s}$$

Vemos que la velocidad entre el centroide del índice 6 del frame 35 representado en azul y el centroide del índice 8 del frame 36 en color verde es de 11.574 cm/s. Finalmente, se validó el algoritmo para medir la velocidad de las burbujas estables y mapeadas entre los frames 35 y 36 en cm/s con 3 decimales, como se muestran en la Tabla 11.

Tabla 11

Velocidad de las burbujas estables entre el par de frames 35 y 36

Frame 35				Frame 36				
Índice	**Área cm²**	**x_1**	**y_1**	**Índice**	**Área cm²**	**x_2**	**y_2**	**Velocidad cm/s**
2	36.00	91.827	20.047	1	40.400	94.973	21.921	8.199
3	28.100	122.775	299.654	4	27.400	121.684	308.582	20.136
4	25.300	210.779	232.860	3	28.200	218.314	239.883	23.062
5	20.600	399.502	392.672	7	20.200	403.461	398.751	16.240
6	18.300	293.527	273.154	8	18.600	296.949	277.029	**11.574**
7	17.400	192.717	39.749	9	18.500	196.224	43.905	12.174
Velocidad promedio en un par de frames								**15.231**

La velocidad promedio es la suma de las velocidades de las burbujas estables, dividida entre número *N* de las burbujas mapeadas y estables de los frames 35 y 36, como se muestra en la Ecuación 8.

$$\bar{V}m = \frac{\sum_{i=1}^{N} Vm_i}{N} \tag{8}$$

Donde:

- $\bar{V}m$ = velocidad promedio (en centímetros por segundo, cm/s);
- Vm_i = velocidades de las burbujas estables (en centímetros por segundo, cm/s);

- i = el índice de la velocidad media;
- N = número de las burbujas mapeadas y estables de los frames 35 y 36.

Según la Ecuación 8, la velocidad promedio entre los frames 35 y 36 es 15.2 cm/s.

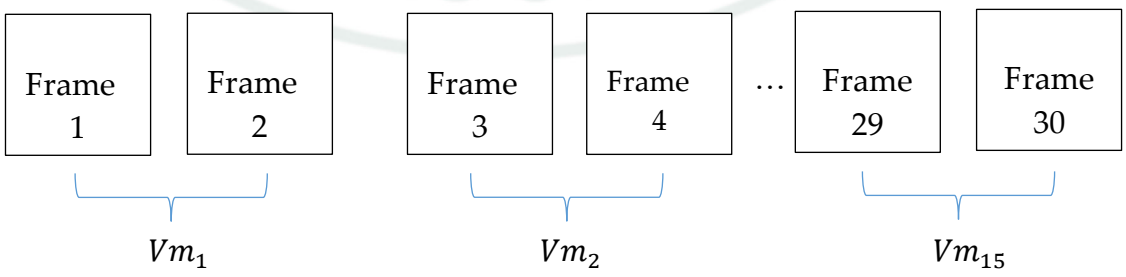
$$\bar{V}m = \frac{(8.199 + 20.136 + 23.062 + 16.240 + \mathbf{11.574} + 12.174)}{6}$$

$$\bar{V}m = 15.231 \frac{\text{cm}}{s}$$

Se muestreó la velocidad promedio de todos los frames en intervalos de un segundo. El video es de 30 fps; es decir, en un segundo hay 30 frames. Cada medida de velocidad se tomó entre un par de frames. Es decir, el frame impar 1 y el frame par 2 fue la primera medida. No se usó el frame par 2 y el frame impar 3 como la segunda medida porque tendríamos 30 medidas en un segundo, haciendo que el tiempo de procesamiento dure más. Por lo tanto, para la segunda medida se tomó el frame impar 3 y el frame par 4, y así sucesivamente. Se concluye que en 30 frames hay 15 pares de frames para 15 velocidades, las cuales se promediaron para obtener la velocidad promedio total en un segundo. La velocidad promedio Vm_1 se tomó entre el frame 1 y el frame 2. Luego, la velocidad promedio Vm_2 se tomo entre el frame 3 y el frame 4. Y así sucesivamente hasta medir la última velocidad promedio Vm_{15} entre el frame 29 y el frame 30, como se muestra en la Figura 36.

Figura 36

Medición de la velocidad promedio total en 1 segundo



Para poder muestrear en el SCADA y el Dashboard IoT la velocidad de la espuma en intervalos de segundos, se determinó la velocidad promedio total Vm_T, como se muestra en la Ecuación 9.

$$Vm_T = \frac{\sum_{i=1}^{n} \bar{V}m_i}{n} \tag{9}$$

Donde:

- Vm_T = velocidad promedio total (en centímetros por segundo, cm/s);
- Vm_i = velocidad promedio de un par de frames (en centímetros por segundo, cm/s);
- i = índice de medida donde i va de 1 a 15;
- n = número de medidas en un segundo. En este caso, n es igual a 15.

En el capítulo de resultados se mostrarán las medidas de la velocidad promedio total en 1 segundo de video y la medición de la velocidad de la espuma durante 7 segundos de video que se implementó en el SCADA y en el Dashboard IoT.

Después de haber obtenido las características dinámicas de la espuma tales como la estabilidad y velocidad de la espuma, se implementaron algoritmos para medir el color de la espuma.

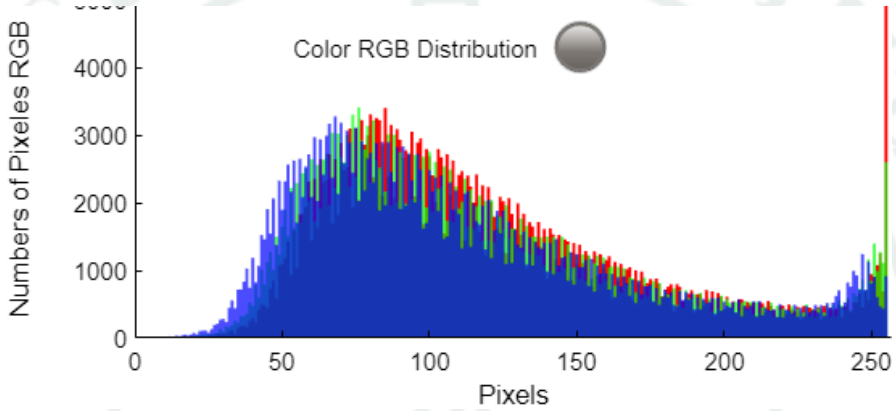
4.4 Algoritmo para medir el color de la espuma

El color de la espuma es un indicador de la cantidad del mineral recuperado u otros componentes que están sobre la superficie de la celda de flotación. Para esta tesis, se implementó el algoritmo del color de la espuma donde se usó el modelo RGB. Luego, se separó la matriz R, G, B del frame número 30 y después se obtuvo el promedio de cada matriz. Para el segundo muestreo se tomó el frame número 60 para sacar el promedio a cada matriz y así sucesivamente. Cada 30 frames equivale a una muestra del promedio de las matrices RGB.

Finalmente, se muestreó el promedio de las matrices de los colores R, G, B cada segundo. Luego se aplicó el histograma de distribución de los colores RGB, como se muestra en la Figura 37. En el capítulo de resultados se explicará el análisis y la interpretación de la importancia de la distribución del color de la espuma.

Figura 37

Histograma de la distribución de los colores R G B



4.5 Algoritmo para medir la textura de la espuma

La matriz GLCM nos da las propiedades de la textura, las cuales son cuatro: contraste, correlación, energía, y homogeneidad. Se puede usar el resultado de la matriz *gray level co-occurrence* (GLCM) para medir las propiedades de la textura de la espuma.

Cuando el flujo de aire aumenta, el área de la burbuja también aumenta, entonces la velocidad o movilidad de la espuma aumenta, lo que resulta en un aumento correspondiente en la tasa de recolección y recuperación de partículas de minerales. Por eso es importante la medición de la energía y la homogeneidad, ya que son directamente proporcionales al flujo de aire. Sin embargo, el contraste es inversamente proporcional. Las imágenes de la espuma con una amplia gama de tamaños, formas y colores de burbujas tienen valores de entropía altos. Los resultados muestran que la intensidad del nivel de gris tiene una correlación positiva con la

recuperación de minerales, por eso es importante medir la correlación, ya que esta mide las dependencias lineales del nivel de gris en la imagen. (Massinaei *et al*., 2019)

Para esta investigación, se obtuvo la *matriz gray-level co-occurrence* (GLCM) de la imagen en escala de grises del frame 30, como se muestra Tabla 12.

Tabla 12

Matriz gray-level co-occurrence GLCM del frame 30

Matriz GLCM							
732	150	0	0	0	0	0	0
151	31479	1926	0	0	0	0	0
0	2049	74872	3960	5	0	0	0
0	3	4095	60580	4493	21	0	0
0	0	6	4591	39156	3683	33	0
0	0	0	38	3717	20975	2531	17
0	0	0	0	41	2554	11322	1624
0	0	0	0	0	13	1638	14605

Luego se obtuvieron sus propiedades tales como contraste, correlación, energía y homogeneidad, como se observa en la Tabla 13.

Tabla 13

Propiedades de la textura de la espuma para el frame 30

Contraste	***Correlación***	***Energía***	***Homogeneidad***
0.1301	0.9753	0.1500	0.9358

El contraste mide la intensidad de contraste entre un píxel y sus vecinos sobre toda la imagen. El contraste también es conocido como varianza o inercia y su rango

es de 0 al *(size(GLCM,1)-1)*^{2}. La correlación mide qué tan correlacionado es un píxel con sus vecinos sobre toda la imagen y su rango es de -1 a +1. La homogeneidad devuelve la medición de la cercanía de la distribución de los elementos de la matriz GLCM a la diagonal de la matriz GLCM y su rango es de 0 a 1.

La energía devuelve la suma de sus elementos al cuadrado y su rango va de 0 a 1, donde 1 es para una imagen constante. La energía es la más importante comparada con los diferentes tipos de textura, ya que muestra una correlación con el tamaño de la burbuja y la presión del aire en la celda de flotación. Por esta razón, en esta investigación se consideró la energía como la textura de la espuma.

La energía también es conocida como uniformidad de energía y segundo momento angular, y se puede representar según la Ecuación 10, donde *p* es el píxel con posición *i,j,* (The MathWorks, 2025).

$$E = \sum_{i,j} p(i,j)^2 \tag{10}$$

Donde:

- E = energía como la textura de la espuma;
- p = píxel con posición *i,j.*

En el capítulo de resultados se medirán la energía como textura de la espuma para 10 intervalos de tiempo en el SCADA y el Dashboard IoT.

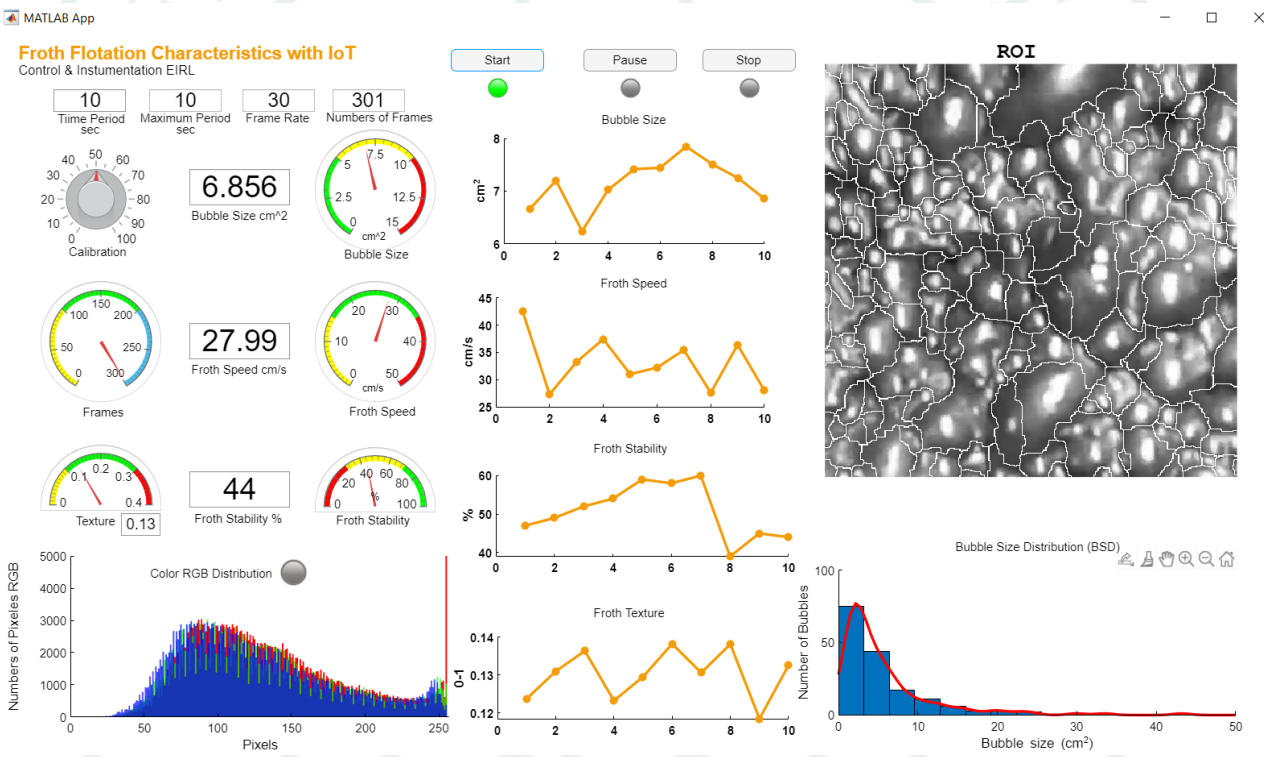
Se pudo implementar el algoritmo para medir las características físicas (tamaño y color), estáticas (textura), y dinámicas (estabilidad y velocidad) de las burbujas de la espuma. El siguiente paso fue la implementación del SCADA y el Dashboard IoT para visualizar y analizar las tendencias de las características de la espuma desde un dispositivo móvil.

5. Interfaz gráfica SCADA con MATLAB App Designer

En esta investigación se implementó una interfaz gráfica o SCADA con MATLAB App Designer para monitorear las características físicas (tamaño y color), estáticas (textura), y dinámicas (estabilidad y velocidad) de la espuma en celdas de flotación, como se muestra en la Figura 38.

Figura 38

SCADA: Froth flotation characteristics



Para la implementación de este SCADA se usó el idioma inglés, ya que los parámetros P&ID en procesos industriales están en inglés. MATLAB App Designer proviene de la versión R2020b, update 6.

MATLAB App Designer contiene varios componentes y cada componente tiene propiedades y funciones *callback*. Cada vez que se presiona un componente o se interactúa con él, llama a su función *callback* para ejecutar un código específico.

Una app puede tener múltiples componentes tales como *button, knob, slider, lamp, edit field, axis,* etc. Estos son componentes de la app. Y estos componentes pueden tener diferentes propiedades, y para acceder a un componente específico, utilizamos la siguiente nomenclatura:

```
app.component.property
```

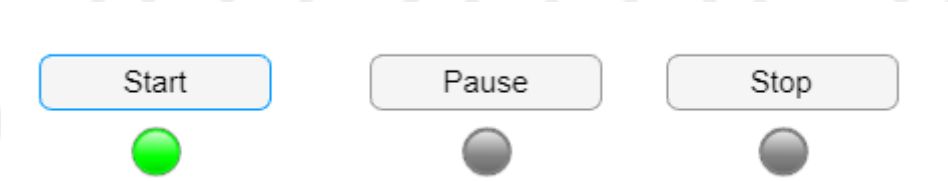
Ahora se explicarán las 9 partes del SCADA *Froth Flotation Characteristics* y los componentes que se agregaron al SCADA.

5.1 Start buttons, pause, stop y lamps

Se agregaron los componentes *start, pause y stop buttons y lamps* para mostrar luz verde, amarilla y roja para el *start*, *pause/resume* y *stop,* respectivamente, como se muestra en la Figura 39.

Figura 39

SCADA: Bottons start, pausa, stop y lamps



Cuando ingresamos un componente de tipo *button* se le asigna el siguiente nombre.

```
app.StartButton
```

Cuando el usuario presiona el botón *start*, la función *callback* del botón *start* comienza a ejecutarse, como se muestra en el Código 1. Primero, da un valor *false* a los *paused button* o *stopped button*; luego, los componentes *lamp2 y lamp3* se ponen de color gris y el componente *lamp* se pone de color verde. Finalmente, ejecuta la función *runProcess*, que es la función principal del algoritmo.

Código 1

Función callback para el start button

Función Callback

```
function StartButtonPushed(app, event)

        app.IsPaused = false;
        app.IsStopped = false;
        app.Lamp.Color = 'g';
        app.Lamp2.Color = '#808080';
        app.Lamp3.Color = '#808080';
        app.runProcess()
    end
```

Cuando se presiona el componente *pause button* se ejecuta el Código 2. Si se presionó el *pause button*, en la propiedad *text* se imprime *resume* para que se visualice en el *button*. Luego se pone en amarillo el *lamp2*. Y si no se presionó, todo permanece igual.

Código 2

Función Callback para el Pause Button

Función Callback

```
function PauseButtonPushed(app, event)

        app.IsPaused = ~app.IsPaused;
        if app.IsPaused
            app.PauseButton.Text = 'Resume';
            app.Lamp.Color = '#808080';
            app.Lamp2.Color = 'y';
            app.Lamp3.Color = '#808080';
        else
        app.PauseButton.Text = 'Pause';
        app.Lamp.Color = 'g';
        app.Lamp2.Color = '#808080';
        app.Lamp3.Color = '#808080';
        end
    end
```

Cuando se presiona el *stop button*, su función *callback* ejecuta el Código 3 para prender el *lamp3* de color rojo.

Código 3

Función callback para el stop button

Función Callback

```
function StopButtonPushed(app, event)
            app.IsStopped = true;
            app.Lamp.Color = '#808080';
            app.Lamp2.Color = '#808080';
            app.Lamp3.Color = 'r';
    end
```

5.2 Numeric edit field para el time period y knob para el H

Se agregó el *numeric edit field* para ingresar en la propiedad *value* el *time period*, que es el tiempo que se desea visualizar en segundos y se almacena en la variable *value*.

Función Callback

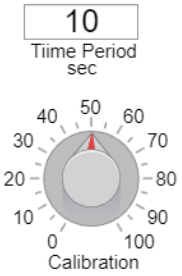
```
function TimePeriodEditFieldValueChanged(app, event)

        value = app.TimePeriodEditField.Value;

    end
```

También, se agregó un *knob* para ingresar el parámetro H de la transformada de máximos extendidos, que va de 0 a 100 como se muestra en la Figura 40. Este componente es parte del programa principal `app.runProcess()`. El nombre del componente es *calibraknob*. La selección deseada en el *knob* se pone en la propiedad *value* y esta se almacena en la variable *cal*.

```
cal = app.CalibraKnob.Value;
```

Figura 40

SCADA: Numeric edit field para el time period y knob para el H

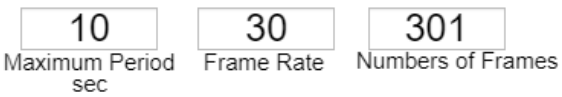


5.3 Edit field para el maximum period, el frame rate y el numbers of frames

Se agregaron los componentes de tipo *numeric edit field* como se muestra en la Figura 41, para visualizar los parámetros tales como el *maximum period* (`app.MaxTimePeriodEditField`), que muestra el máximo tiempo que se puede visualizar del video con formato mp4; el *frame rate* (`app.FrameRateEditField`), que es el número de frames que hay en un segundo, también llamado el fps del video; y el *numbers of frames* (`app.NroFramesEditField`), que es el número de frames que hay en el video.

Figura 41

SCADA: Maximum period, frame rate y numbers of frames

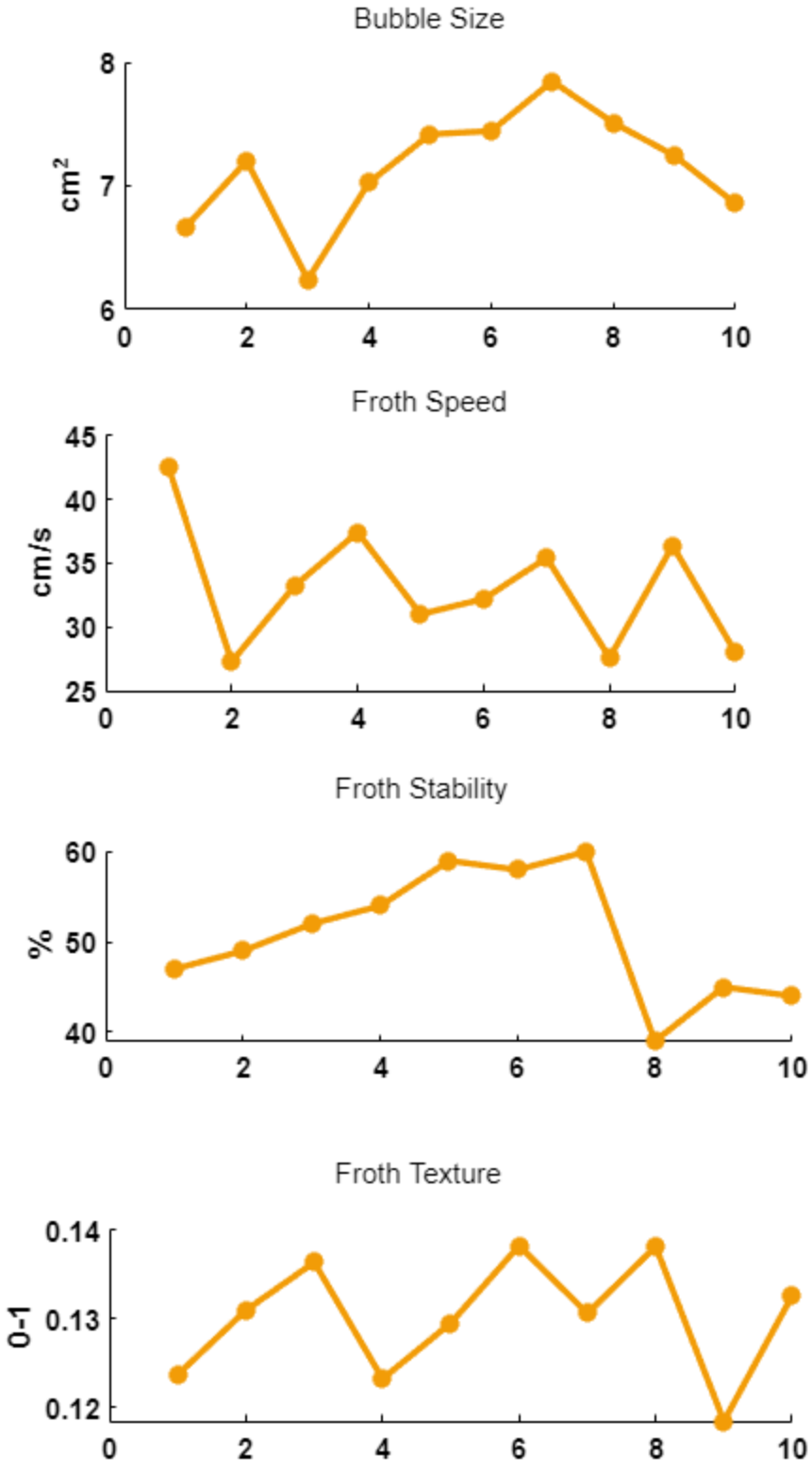


5.4 Axes para las tendencias de las características de la espuma

Se usaron los componentes te tipo *axes* como se muestran Figura 42.

Figura 42

SCADA: Axes para las tendencias de las características de la espuma



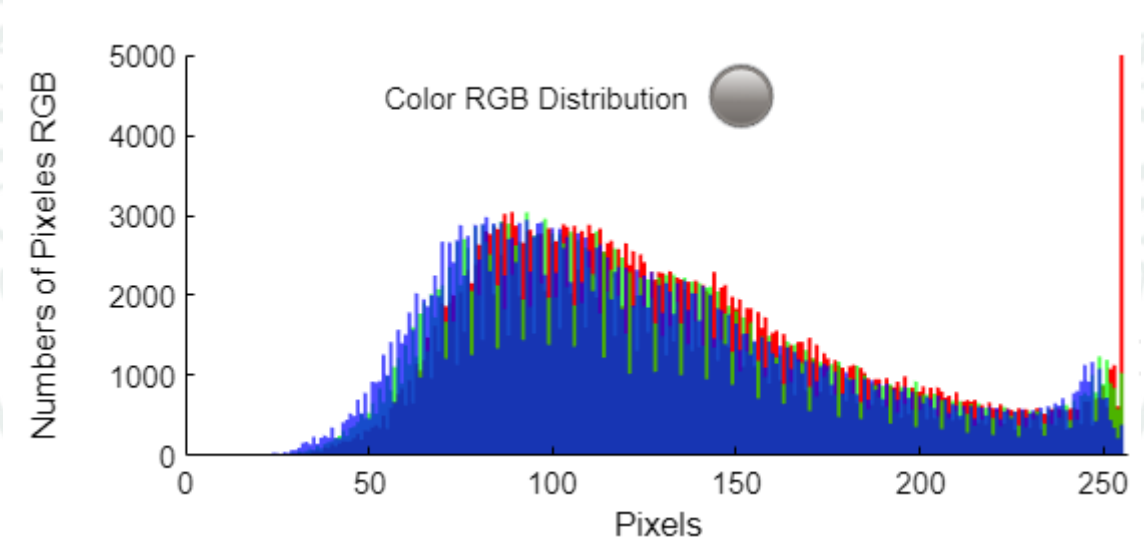
La componente para las tendencias del *bubble size* o tamaño de la burbuja en cm^2 es (`app.UIAxesSize`); para el *froth speed* o velocidad de la espuma en cm/s es (`app.UIAxesSpeed`); para el *froth stability* o estabilidad de la espuma en porcentaje es (`app.UIAxesStability`); y para el *texture* o textura de tipo energía de 0 a 1 es (`app.UIAxesTexture`).

5.5 Histogramas para las tres componentes de color RGB.

Se agregó el componente *axes* (`app.UIAxesRGB`), para los histogramas para las tres componentes RGB, como se muestra en la Figura 43.

Figura 43

SCADA: Histogramas para las tres componentes de color RGB

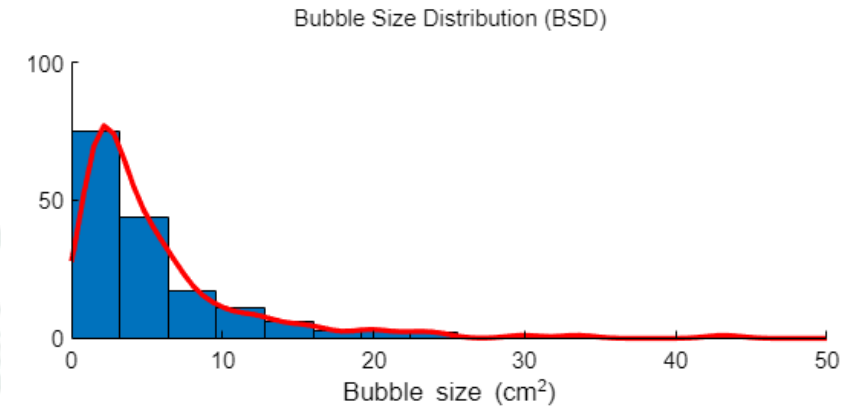


5.6 Histograma para el BSD

Se agregó el componente *axes*, (app.UIAxesBSD), para el histograma donde se mostrará la distribución del tamaño de las burbujas (BSD), como se muestra en la Figura 44.

Figura 44

SCADA: Histograma para el BSD

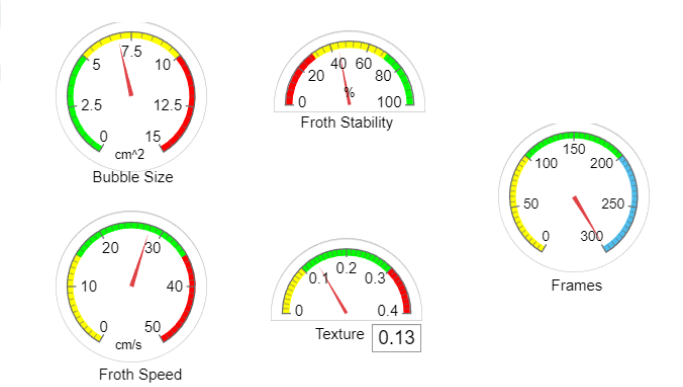


5.7 Widgets para las características de la espuma y el progreso del algoritmo

Se agregaron widgets de tipo *gauges* circulares y semicirculares (app.SizeGauge, app.SpeedGauge, app.StabilityGauge, app.TextureGauge, app.FramesGauge), como se muestra en la Figura 45, para visualizar las características de la espuma y para ver el progreso del algoritmo basado en los frames.

Figura 45

Widgets para las características de la espuma y los frames

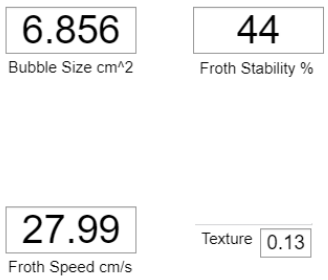


5.8 *Edit field* para las características numéricas de la espuma

Se agregaron los componentes de tipo *numeric edit field* (app.SizeEditField, app.SpeedEditField, app.StabilityEditField, app.TextureEditField) como se muestra en la Figura 46, para visualizar de forma numérica las características de la espuma.

Figura 46

SCADA: Edit field para las características numéricas de la espuma

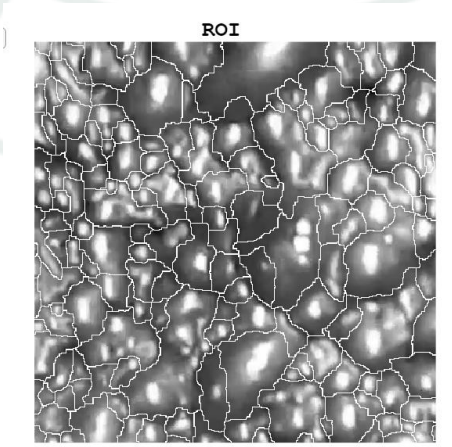


5.9 Axes para el ROI con el área de las burbujas

Se agregó el componente Axes (`app.UIAxesFroth`) para cargar las líneas *watershed* superpuestas sobre la imagen en escala de grises del frame par para determinar el ROI, como se muestra Figura 47.

Figura 47

SCADA: Axes para el ROI con el área de las burbujas

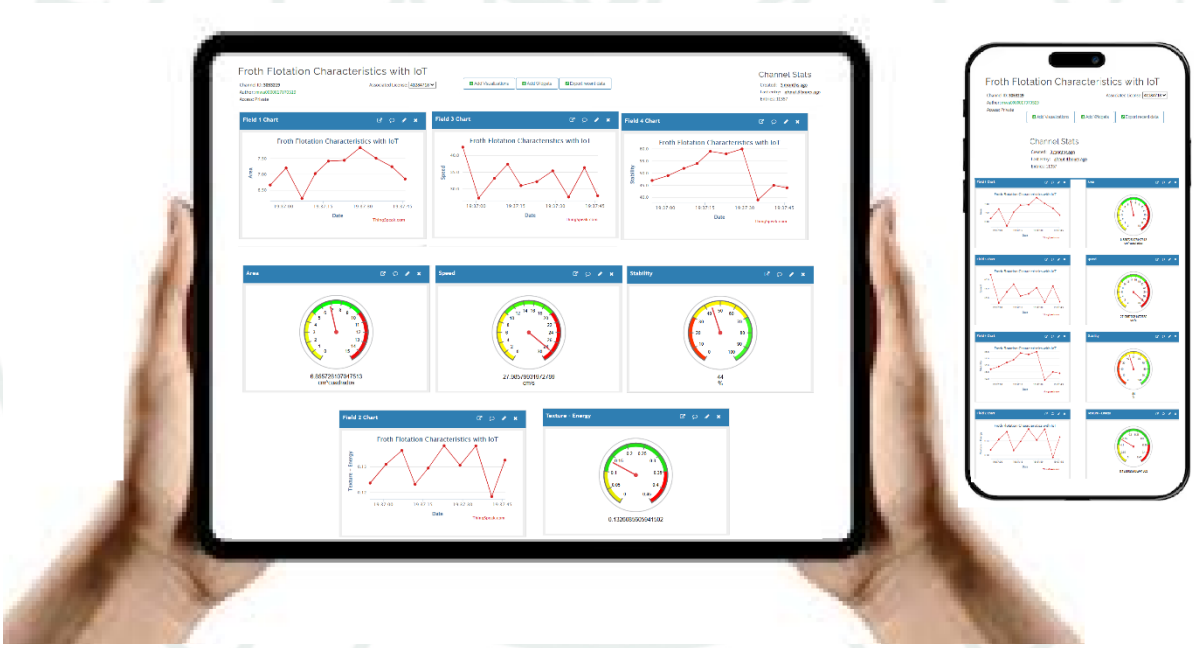


6. Dashboard IoT con ThingSpeak™

En esta investigación se usó ThingSpeak™ del año 2025. Esta plataforma no tiene versión porque está en la nube y se actualiza automáticamente. Se usó esta plataforma para visualizar y analizar el flujo de datos de las características de la espuma desde la nube. En la Figura 48, se muestran las tendencias de las características de la espuma en dispositivos tales como *tablets* o teléfonos móviles conectados a internet para acceder a la plataforma IoT de ThingSpeak™ usando los protocolos REST, API, o MQT.

Figura 48

Visualización de las características de la espuma con ThingSpeak™



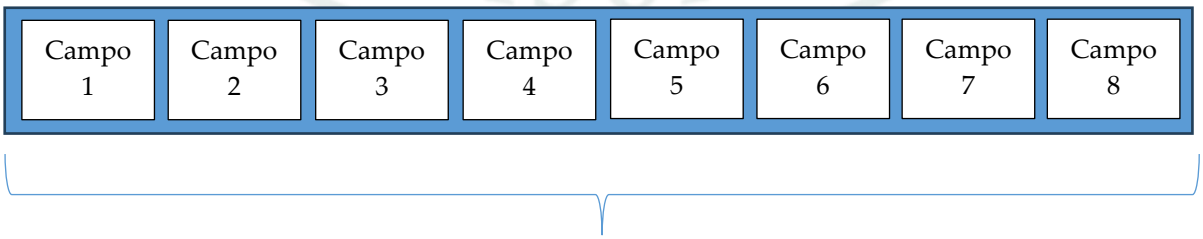
ThingSpeak™ está en la página web https://ThingSpeakTM.mathworks.com/. Con ThingSpeak™ se puede escribir datos provenientes desde dispositivos hacia un canal ThingSpeak. Mientras más canales haya, más dispositivos se pueden conectar. Se puede escribir un mensaje a un canal ThingSpeak™ en el cual se puede tener hasta 8 campos para variables o datos como temperatura, humedad, etc., y cada mensaje no

debe exceder más de 3000 bytes, como se muestra en la Figura 49. Se puede escribir un mensaje a un canal ThingSpeak™ usando los protocolos REST, API y MQTT. También se puede escribir a un canal desde MATLAB® usando el comando *thingSpeakwrite* o desde un canal ThingSpeak™ usando MATLAB Analytics o MATLAB Visualization Apps. Y se puede realizar una escritura a un canal de imagen ThingSpeak™, pero esto consume 100 mensajes. ThingSpeak™ vende licencias anuales por Unidades, donde en una unidad se puede escribir hasta 33 millones de mensajes. Una unidad provee la habilidad de tener un número determinado de canales dependiendo del tipo de licencia. Una licencia estándar para una unidad provee hasta 250 canales. Estas Unidades pueden durar desde 365 a 395 días según la fecha de compra, ya que la licencia vence al final del mes.

ThingSpeak alerta cuando se están acabando los mensajes para poder comprar más unidades. Se puede almacenar los datos históricos hasta 10 millones de mensajes por tres años con licencia gratis, y si se paga por una licencia, se puede almacenar hasta 100 millones de mensajes históricos por 3 años. Si el canal se excede de mensajes, ya no podrá recibir más mensajes. Si se desea almacenar por más tiempo, se puede hacer solicitándolo por email. Para acceder a ThingSpeak™, se puede usar una cuenta gratuita que permite enviar datos con intervalos de 15 segundos. (The MathWorks, Inc., 2025c)

Figura 49

Características de un mensaje ThingSpeak



Para esta investigación, inicialmente se planteó muestrear las variables con intervalos de 1 segundo y para ello se tuvo que comprar una licencia estudiantil. Esta licencia incluye 1 Unidad con una capacidad máxima de 10 canales para una capacidad máxima de 33 millones de mensajes por año o 90,000 mensajes por día. Cada canal puede tener hasta 8 campos para las variables, de los cuales solo se usarán 4 campos para la velocidad, el área, la textura y la estabilidad. Los 4 campos sobrantes quedarían vacíos. El costo de la licencia por un año fue de $45.00. Se puede enviar imágenes de hasta 5MB. Cada imagen usa 200 mensajes. Se puede compartir el canal en forma ilimitada. En la Figura 50 se muestran las características de esta licencia.

Figura 50

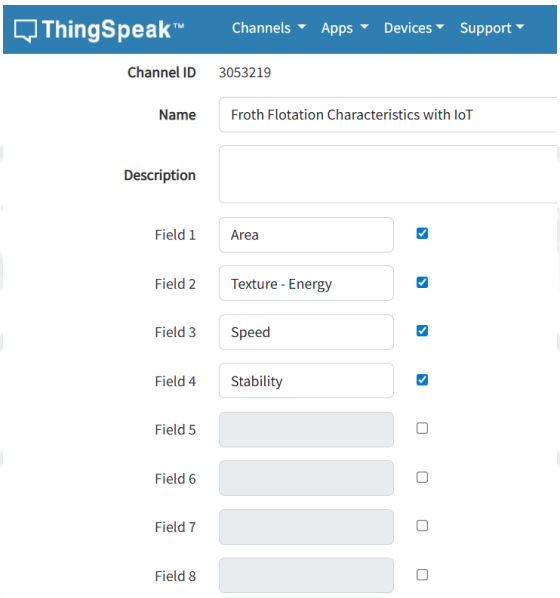
Características de la Licencia ThingSpeak *para estudiantes*

	FREE For small non-commercial projects	STUDENT For students at degree-granting institutions[1]
Scalable for larger projects	✖ No. Annual usage is capped.	✔
Number of messages	3 million/year (~8,200/day)[2]	33 million/year per unit (~90,000/day per unit)[2]
Message update interval limit	Every 15 seconds	Every second
Number of channels	4	10 per unit
MATLAB Compute Timeout	20 seconds	20 seconds
Private channel sharing	Limited to 3 shares	Unlimited
Technical Support	Community Support	Community Support
Max image size	✖ Image feature unavailable	5 MB
Messages used per image	✖	100

Se creó un nuevo canal para la tesis con el nombre *Froth Flotation Characteristics with IoT*, con número ID 3053219. En este canal se usaron 4 campos donde se escribirán los datos de las variables velocidad, tamaño, textura y estabilidad, respectivamente, como se muestra en la Figura 51.

Figura 51

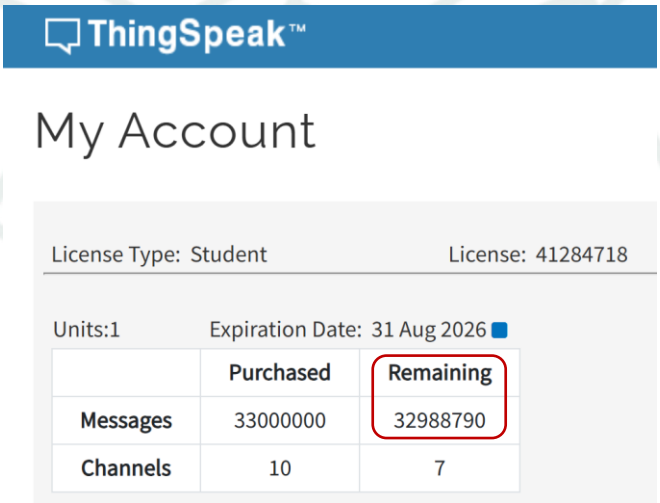
Campos del canal ThingSpeak *para los parámetros de la espuma*



En la Figura 52 se muestra la capacidad de mensajes que se adquirió con la licencia: 33 millones de mensajes y 10 canales. Se fueron usando mensajes para realizar las pruebas y ahora quedan 32,988,790 mensajes.

Figura 52

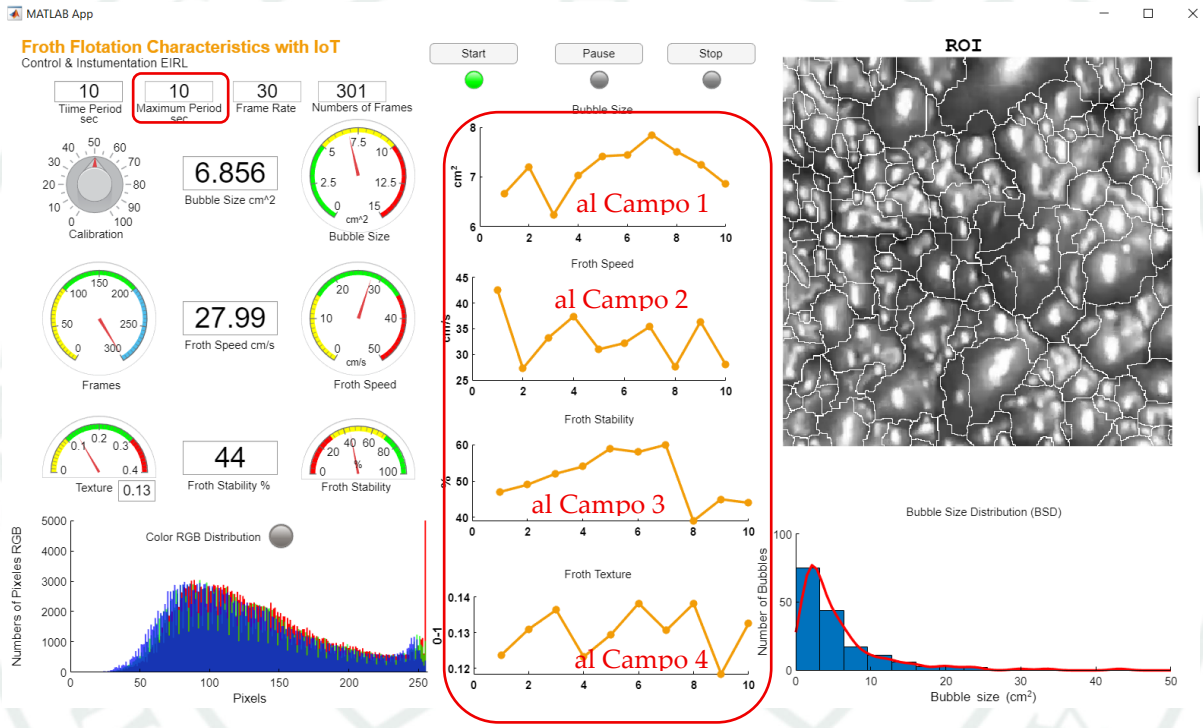
Capacidad de mensajes comprados y mensajes sobrantes



Los mensajes se escriben en los campos de un canal. Para comprobar esto, vemos que en la Figura 53 se muestra el SCADA hecho en MATLAB App Designer, donde se grafican las características de la espuma en intervalos (una muestra por intervalo).

Figura 53

Escritura de las características en campos de un canal ThingSpeak



En el Código 4, vemos que se enviaron mensajes que empaquetan los datos de la velocidad (VelSamp), área (AreaSamp), textura (TextSamp) y estabilidad (StabSamp) durante un periodo de 10 intervalos para escribirlos en los cuatro campos del Canal de ThingSpeak con ID: 3053219, donde *writekey* es el código de permiso para el canal.

Código 4

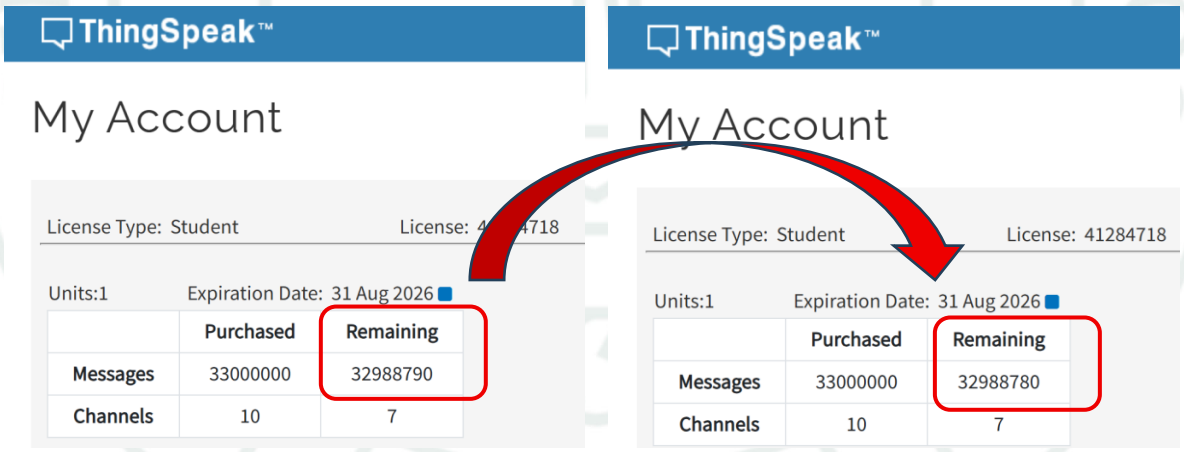
Escritura de datos en un canal IoT de ThingSpeak *desde* MATLAB App Designer

```
ThingSpeakTMWrite(3053219,
'Fields',[1,2,3,4],
'Values',[AreaSamp(z,1),TextSamp(z,1),VelSamp(z,1),StabSamp(z,1)],
'WriteKey','XXXXXXXXXXXXXXX')
```

Se demostró que un mensaje contiene todos los campos de un canal porque anteriormente teníamos 32,988,790, y si restamos 10 mensajes donde se escribieron las características de la espuma en 4 campos durante 10 intervalos, ahora nos quedaría 32,988,780, como se muestra en la Figura 54.

Figura 54

Envió de 10 mensajes con sus campos en 10 intervalos



En la Figura 55, se muestran los 10 mensajes que se escribieron al canal ThingSpeak con ID 3053219 y con nombre *Froth Flotation Characteristics with IoT* y vemos que no se perdió ningún dato y que es igual a las tendencias del SCADA de la Figura 53.

Figura 55

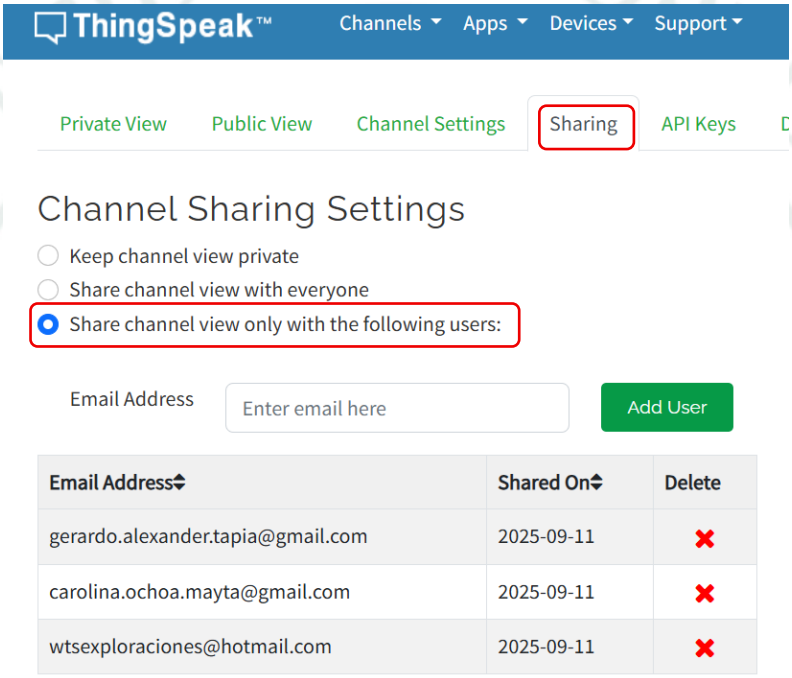
Canal ThingSpeak: *Froth Flotation Characteristics with IoT*



Como se puede ver en la Figura 55, el canal es privado, pero se puede compartir a usuarios por correo electrónico o se puede hacer público seleccionando la opción: *share channel view with everyone*, como se muestra en la Figura 56.

Figura 56

Configuración del canal a modo público



Si se configura el canal a modo público, solo se necesita el User ID: mwa0000017070519. En privado se ven los últimos datos recientes que se han configurado, mientras que en modo público se muestran todos los datos, y si se desean ver los últimos datos recientes, entonces cada usuario tendría que configurarlo para visualizar los últimos datos deseados.

Se hizo un análisis para saber cuántos mensajes se enviarían en un año si el intervalo de tiempo fuera de 1 segundo. En la Tabla 14, se muestra la cantidad de mensajes que se enviarían para 4 campos activos las 24 horas al día, los 365 días al año y el número de mensajes sobrantes. El número de mensajes que se enviarían por minuto es 60, por hora 3,600, por día 86,400, por mes 10,512,000, y por año 31, 536,000.

Entonces, no habría problemas porque la capacidad máxima es de 33 millones al año. Por lo tanto, nos sobraría 1,464,000 antes de que termine la licencia. Se puede usar 4 variables más sin ningún problema ya que sobran 4 campos y cada mensaje contiene hasta 8 campos. Pero si se quisiera enviar más de 8 campos, se tendría que comprar otra unidad. También se podría enviar mensajes en intervalos de 5, 10, 15 o más segundos por lo que sobrarían más mensajes y por lo tanto se podrían usar hasta 10 canales. Si se desea enviar los datos para controlar la velocidad de la espuma y el nivel de la espuma, es recomendable muestrear en intervalos de 5 segundos debido al tiempo computacional de los algoritmos calculados en los resultados del Capítulo V.

Tabla 14

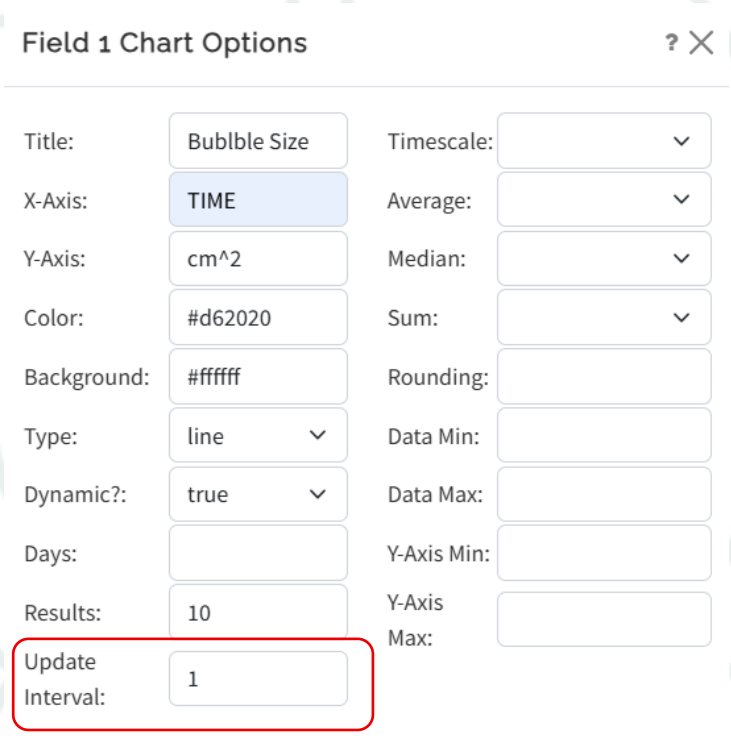
Cálculo de mensajes que se pueden enviar en ThingSpeak

Unidad	Canal	Campos	Variables	Mensajes Por segundo	Mensajes por hora	Mensajes por día	Mensajes por mes	Mensajes por año
Capacidad de la Unidad que se compró								
1	1-10	1-8	8	1	3,750	90,000	2,750,000	**33,000,000**
Mensajes que se utilizaran por hora, día, mes y año en esta tesis								
1	1	1	Área	1	3,600	86,400	2,628,000	31,536,000
		2	Textura					
		3	Velocidad					
		4	Estabilidad					
Total de mensajes sobrantes								1,464,000

Se puede configurar en el canal ThingSpeak™ para un intervalo de actualización de un segundo haciendo clic en *chart options*, como se muestra en la Figura 57, pero se configuró para un intervalo de actualización de cinco segundos debido al tiempo computacional de los algoritmos. También se puede configurar en *Chart Options* los títulos de los *axes*, el color, el fondo, el tipo de gráfico, los últimos resultados que se desean mostrar, etc.

Figura 57

Configuración del tiempo del intervalo de actualización a 1 segundo

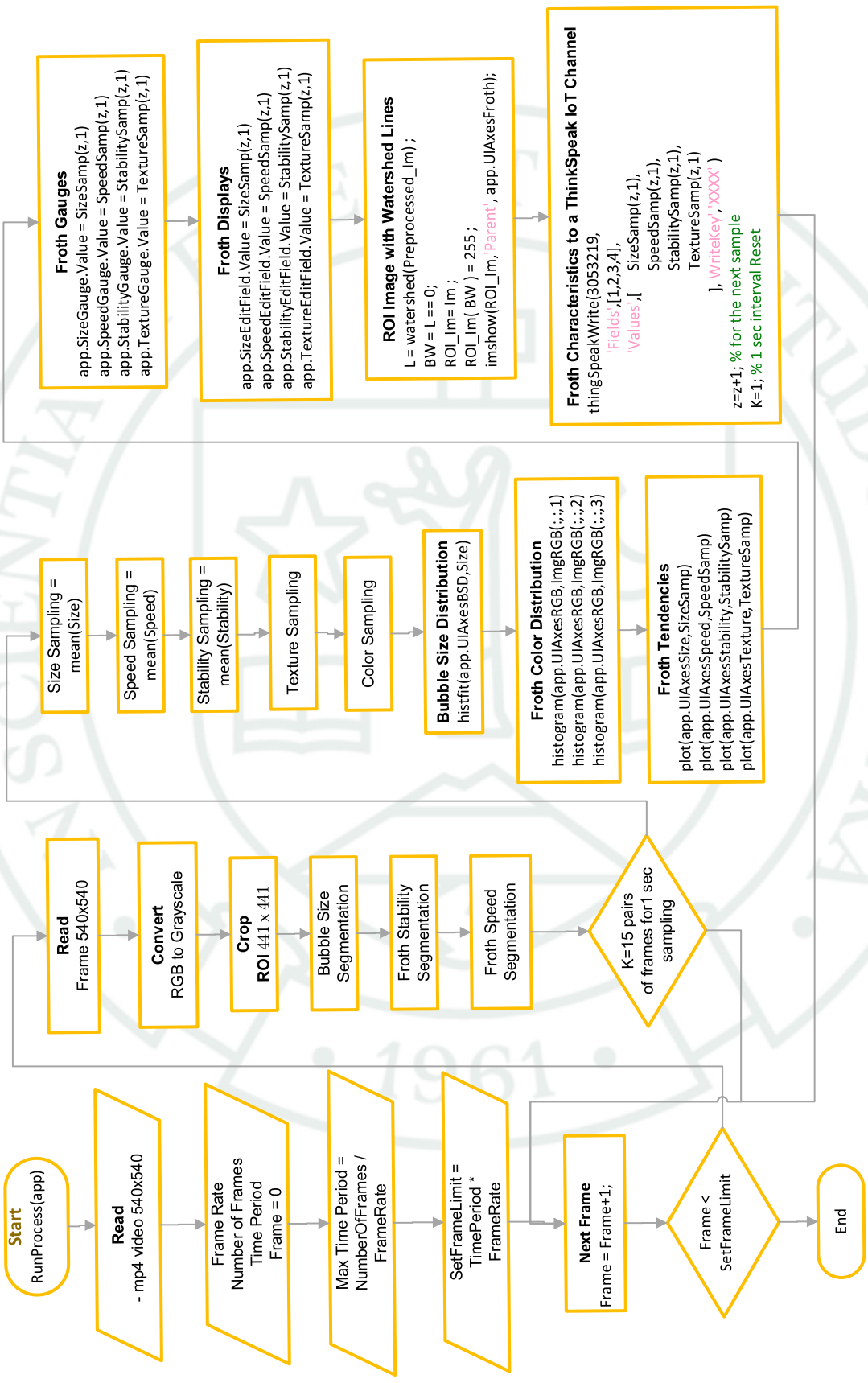


7. Diagrama de flujo para el SCADA y el Dashboard IoT

En la Figura 58, se muestra el diagrama de flujo para el SCADA y el Dashboard IoT para medir las características de la espuma.

Figura 58

Diagrama de flujo para el SCADA y el Dashboard IoT



Ahora se procederá a explicar cada bloque del algoritmo. Después de presionar el botón *start* como se muestra en la Figura 39 y el Código 1, se inicia el algoritmo con la función *RunProcess(app)*. Luego, se lee el video mp4, 540x540 píxeles y se obtiene el *frame rate*, *number of frames*, y se ingresa el *time period*, tiempo que correrá el algoritmo. Se calcula el *maximum time period* para saber cuánto tiempo puede correr el algoritmo para dicho video. Se calcula el *SetFrameLimit* para saber cuál es el último frame par del video que correrá el algoritmo. El *while loop* correrá desde el frame 1 hasta el *SetFrameLimit*. Luego se lee el frame 1, 540 × 540 píxeles, para convertirlo en escala de grises y luego esta es recortada a 441 × 441 para obtener la región de interés (ROI). Al ROI se le aplica el algoritmo de segmentación para medir el tamaño de las burbujas en cm^2, como se muestra en la Figura 14.

Se vuelve a aplicar el mismo algoritmo para el siguiente frame par. Se promedian las áreas de las burbujas del frame 1 con las áreas del siguiente frame par. Luego se aplica el algoritmo de segmentación para medir la estabilidad en porcentaje, como se muestra en el diagrama de flujo de la Figura 30, donde se logra medir la estabilidad en el siguiente frame par.

Luego se aplica el algoritmo de segmentación para medir la velocidad promedio de la espuma en cm/s en un par de frames, como se muestra en la Ecuación 8. Luego se muestrearon las características de la espuma cada segundo de video. Debido a que en cada par de frames se mide la velocidad y la estabilidad, entonces, como el video es de 30 fps, en 1 segundo tendríamos 15 pares de medidas. Por lo tanto, cuando k sea igual a 15, se tomará el promedio de todas las medidas para lograr una muestra promedio cada segundo de video. Entonces, en cada segundo se obtiene el muestreo del promedio del área de las burbujas, de la estabilidad y de la velocidad. También, se obtiene cada 30 frames el color y la textura de la espuma. Luego, por medio de un histograma, se obtiene la distribución del tamaño de las burbujas (BSD) y la distribución del color de la espuma (FCD).

Finalmente, se agregan los *axis* para las tendencias, los *displays* y los *gauges* para las características de la espuma. Luego se agrega el *axis* para las líneas *watershed* interpuestas sobre la imagen del ROI para ver en forma de video el movimiento de la espuma. La última parte del algoritmo es la creación del Dashboard IoT, y para ello, se escriben las principales características de la espuma en los campos del canal ThingSpeak™ llamado *Froth Flotation Characteristics with IoT*. Para que otro pueda ver las medidas del Dashboard IoT, debe tener una cuenta ThingSpeak™ o ser invitado por el creador del canal por correo electrónico ya que es un canal privado.

CAPÍTULO V: PRUEBAS, RESULTADOS Y DISCUSIONES

1. Validación del BSD propuesto

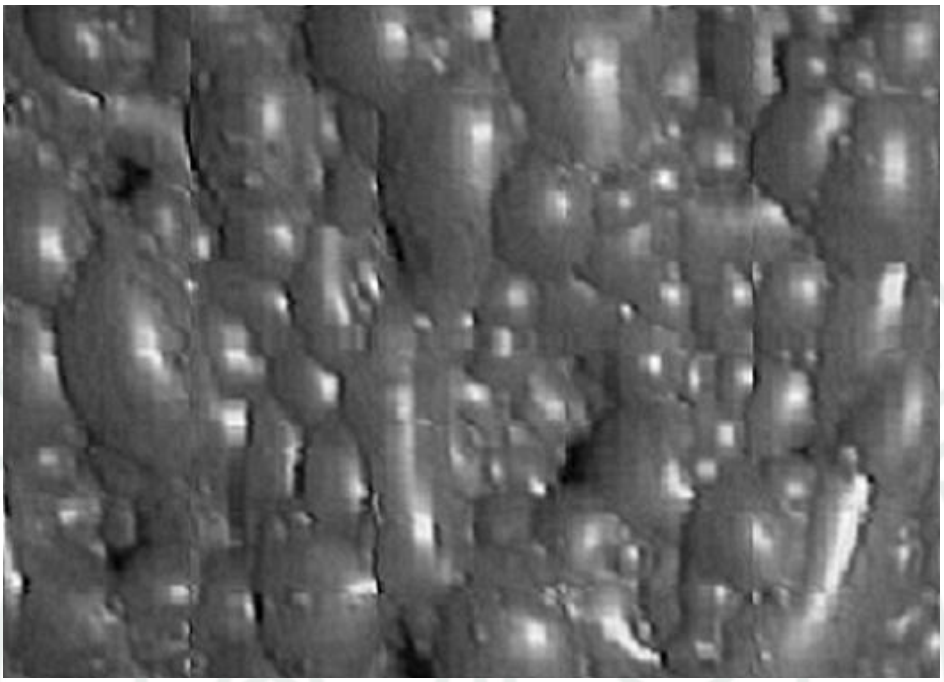
La validación del BSD obtenido mediante el método de segmentación propuesto se realizó mediante su comparación con el BSD de referencia propuesto por Wright (1999). Para cuantificar las diferencias entre ambas distribuciones, se calcularon las métricas MAE, MSE y RMSE. Asimismo, se empleó la distancia de Wasserstein (EMD), la cual permite evaluar la disimilitud global entre distribuciones al considerar tanto la magnitud como el desplazamiento de sus frecuencias.

1.1 Comparación del BSD de referencia y el BSD propuesto

Wright (1999), en su tesis *The Development of a Vision-Based Flotation Froth Analysis System*, realizó una segmentación manual de las burbujas para verificar la exactitud del algoritmo automático diseñado para medir su área. Debido a que la segmentación manual proporciona una referencia exacta, se adoptó dicho procedimiento y su imagen patrón para validar la métrica de la distribución del tamaño de las burbujas (BSD, por sus siglas en inglés) en esta investigación. En la Figura 59, se presenta la imagen denominada "Froth image I1" de 640 × 480 píxeles capturada por el autor, la cual se empleará como el estándar de comparación para la validación del BSD propuesto.

Figura 59

Imagen de referencia (ground truth)

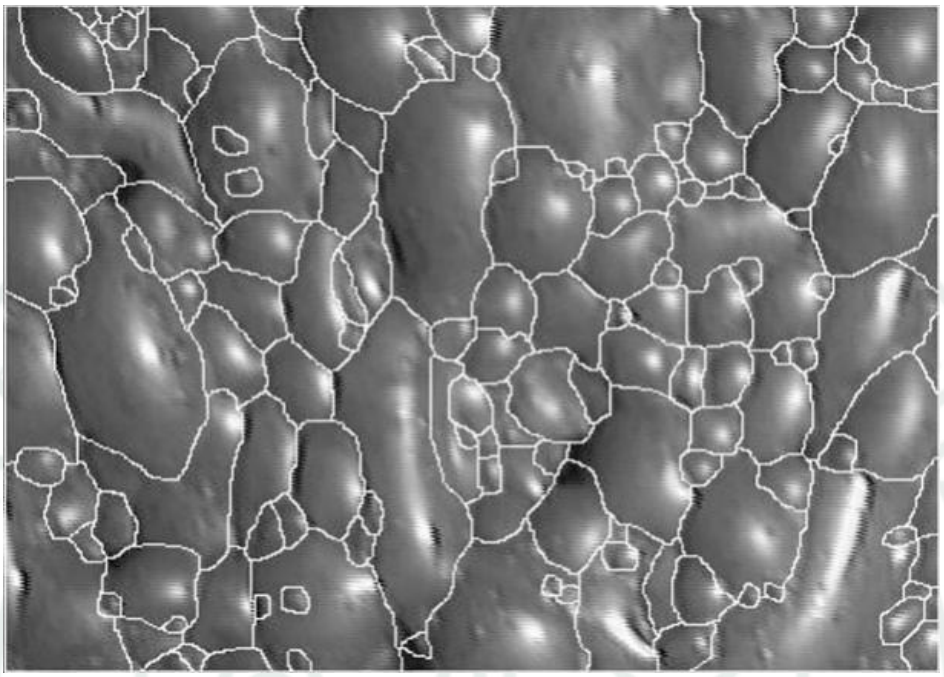


Nota. Reproducido de "Froth image I1", por B. A. Wright, 1999, *The Development of a vision-based flotation froth analysis system* [Tesis de maestría, University of Cape Town], OpenUCT Institutional Repository. http://hdl.handle.net/11427/9923

Y en la Figura 60, se presenta la imagen patrón denominada "Froth image I1", la cual fue segmentada manualmente por Wright (1999), quien la denominó posteriormente "Manual segmentation of froth image I1".

Figura 60

Imagen segmentada manualmente

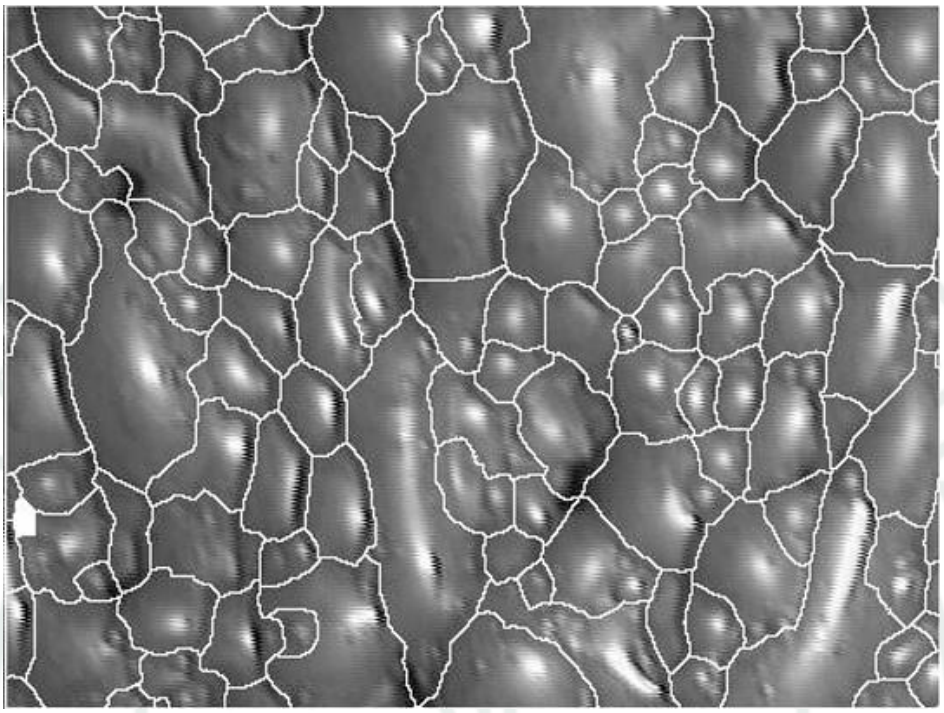


Nota. Reproducido de "Manual segmentation of froth image I1", por B. A. Wright, 1999, *The Development of a vision-based flotation froth analysis system* [Tesis de maestría, University of Cape Town], OpenUCT Institutional Repository. http://hdl.handle.net/11427/9923

En la Figura 61, se presenta la segmentación automática con *fast watershed* implementada por Wright (1999).

Figura 61

Imagen segmentada automáticamente con fast watershed (método A)

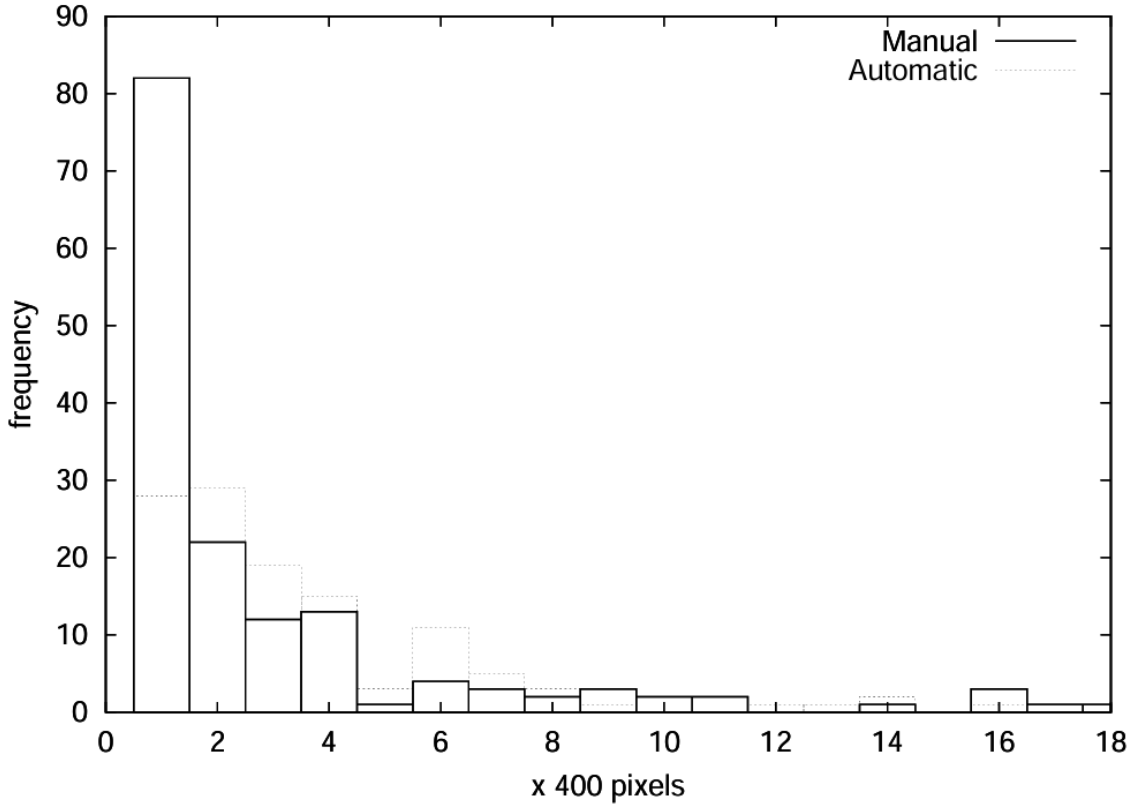


Nota. Reproducido de "Automatic segmentation of froth image I1", por B. A. Wright, 1999, *The Development of a vision-based flotation froth analysis system* [Tesis de maestría, University of Cape Town], OpenUCT Institutional Repository. http://hdl.handle.net/11427/9923

En la Figura 62, se muestra el BSD denominado "Manual and automatic distributions for bubble area-image I1" desarrollado por Wright (1999). Las líneas continuas negras indican la segmentación manual y las líneas punteadas indican la segmentación automática.

Figura 62

BSD de la segmentación manual y automática

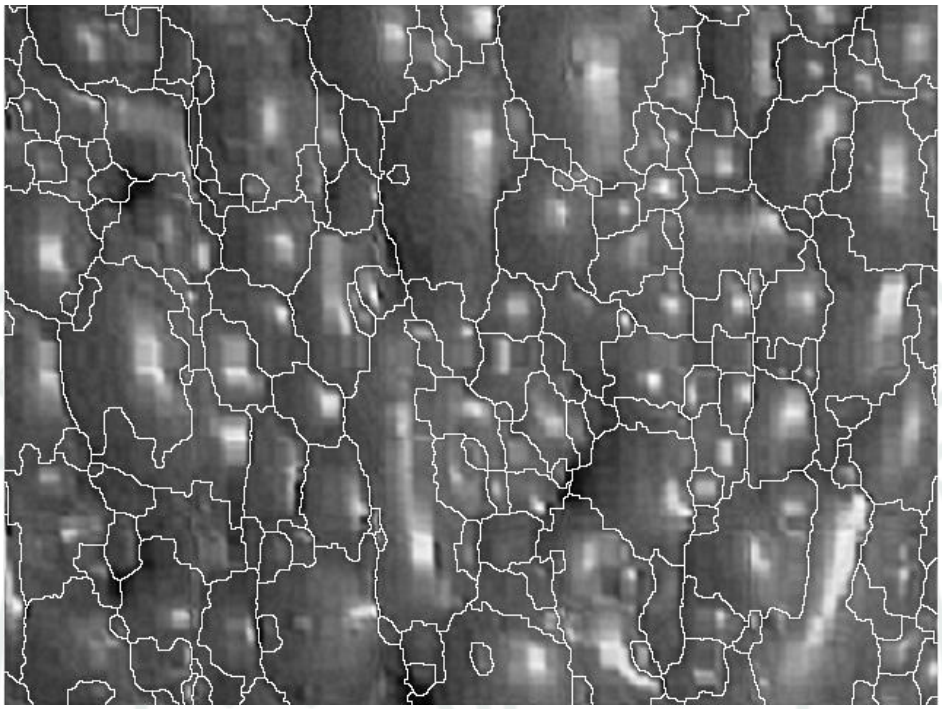


Nota. Reproducido de "Manual and automatic distributions for bubble area-image I1", por B. A. Wright, 1999, *The Development of a vision-based flotation froth analysis system* [Tesis de maestría, University of Cape Town], OpenUCT Institutional Repository. http://hdl.handle.net/11427/9923

Por otro lado, en la Figura 63, se muestra la segmentación de la imagen "Froth image I1" de Wright (1999) basada en el algoritmo *marker-controlled watershed* implementado en esta investigación.

Figura 63

Segmentación de la imagen de referencia basada en el método B propuesto



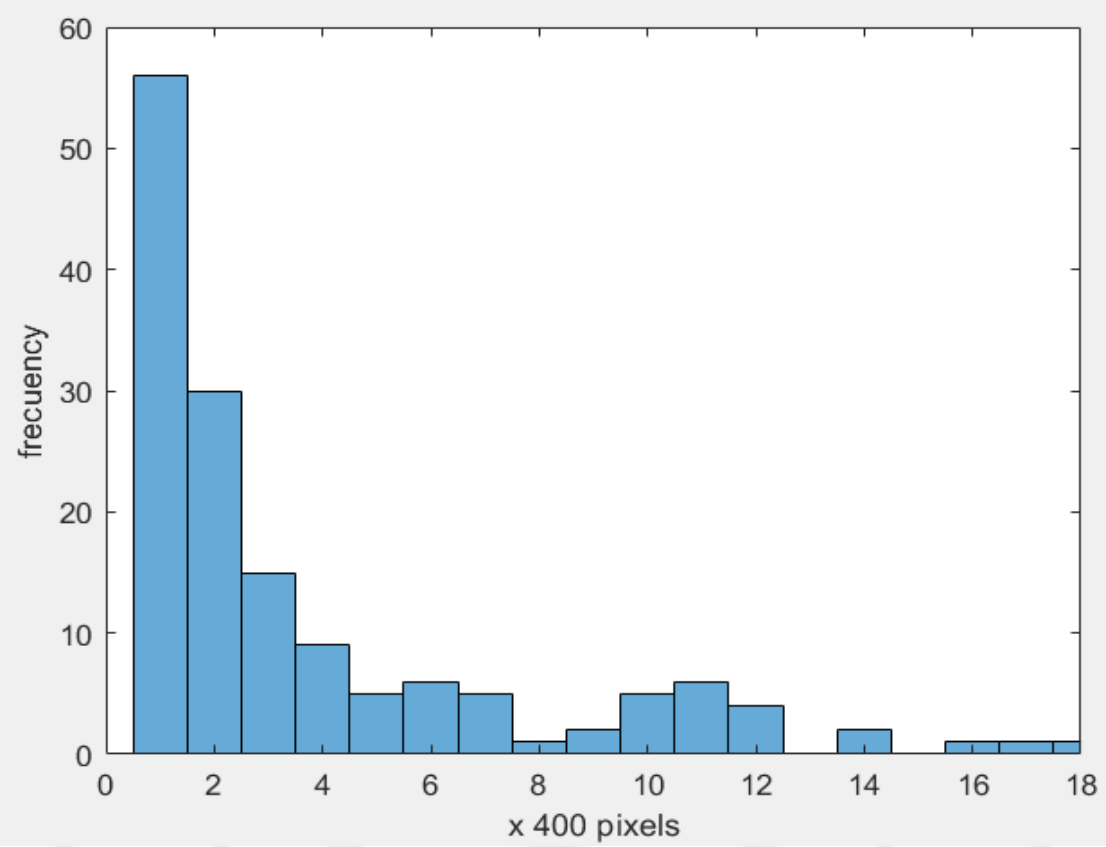
Nota. Adaptada de "Froth image I1", por B. A. Wright, 1999, *The Development of a vision-based flotation froth analysis system* [Tesis de maestría, University of Cape Town], OpenUCT Institutional Repository. http://hdl.handle.net/11427/9923

Debido a que las mediciones realizadas por Wright (1999) se presentan en píxeles, se omitió el factor de escalamiento en centímetros cuadrados (cm^2) en el algoritmo desarrollado para esta investigación, permitiendo así una comparación directa bajo la misma unidad de medida. En la Figura 64 se muestra el BSD obtenido mediante el método propuesto basado en el algoritmo *marker-controlled watershed*. En dicho histograma o BSD, el eje *y* representa la frecuencia del número de burbujas, mientras que el eje *x* indica el área de estas en píxeles. Los *bins* del histograma comienzan en 0.5 con un ancho constante de 400 px. De este modo, el primer intervalo

abarca un rango de 0.5 × 400 px a 1.5 × 400 px (de 200 a 600 px con un punto medio de 400 px); este patrón se repite sucesivamente hasta el último intervalo, cuyo rango comprende de 17.5 × 400 px a 18.5 × 400 px (de 7000 a 7400 px con un punto medio de 7200 px).

Figura 64

BSD de la imagen patrón obtenido en el algoritmo de esta investigación



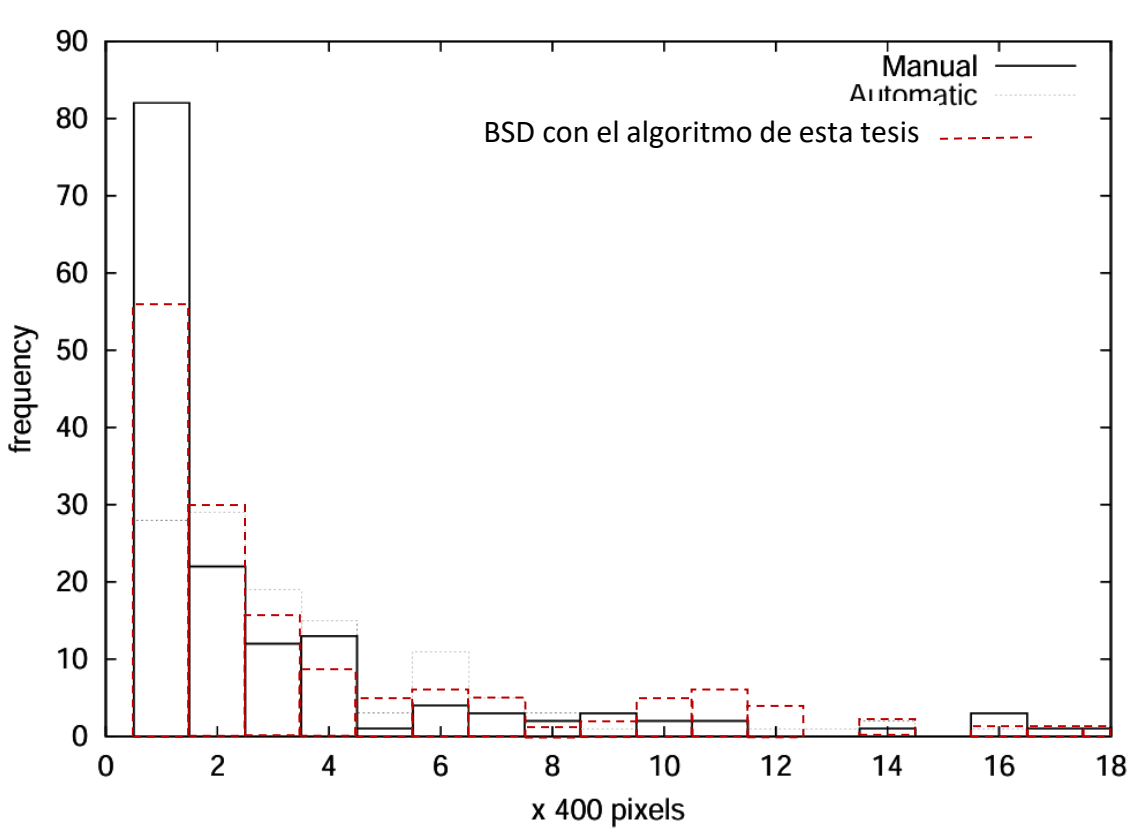
Nota. Todos los *bins* poseen un ancho constante de 400 px. El primer *bin* abarca un rango de 200 a 600 px, con su punto medio situado en los 400 px.

En la Figura 65, se realiza la comparación mediante la superposición del BSD obtenido con el algoritmo de esta investigación y los BSD de las segmentaciones manual y automática desarrolladas por Wright (1999). Las líneas negras continuas representan el BSD de la segmentación manual, mientras que las líneas punteadas

negras corresponden a la segmentación automática de Wright. Las líneas discontinuas en rojo corresponden al BSD generado por el método de segmentación propuesto basado en el algoritmo *watershed* controlado por marcadores (*marker-controlled watershed*).

Figura 65

Comparación de los BSD de la segmentación manual y automática con el método de segmentación propuesto



Nota. Adaptado de "Manual and automatic distributions for bubble area-image I1", por B. A. Wright, 1999, *The Development of a vision-based flotation froth analysis system*

[Tesis de maestría, University of Cape Town], OpenUCT Institutional Repository. http://hdl.handle.net/11427/9923.

1.2 Resultados de las métricas de error para los BSD

Para determinar la magnitud del error en las mediciones de los BSD de la Figura 65, se cuantificaron las discrepancias de los histogramas bin por bin mediante el error absoluto medio (MAE), el error cuadrático medio (MSE) y la raíz del error cuadrático medio (RMSE); asimismo, se complementó el análisis con la distancia de Wasserstein (EMD) para evaluar la disimilitud entre las distribuciones. Para ello, se consideró la frecuencia de referencia *ground truth* o número real de burbujas (fr) proveniente del método de segmentación manual desarrollado por Wright (1999). De igual manera, se consideró como frecuencia medida (fm_A) el número de burbujas registrado en el BSD generado por el método de segmentación A, basado en el algoritmo *fast watershed* implementado por Wright. Finalmente, se consideró como frecuencia medida (fm_B) al número de burbujas correspondiente al BSD obtenido mediante el método de segmentación B propuesto basado en el algoritmo *marker-controlled watershed*.

Para el cálculo de las métricas de error también fue necesario considerar la posición de los contenedores (*bins*) del histograma. En este estudio, los *bins* representan categorías de tamaño de burbuja expresadas en píxeles, por lo que no constituyen una unidad física independiente. Los histogramas evaluados estuvieron conformados por 18 bins, cada uno con un ancho de 400 px. El primer bin abarca el intervalo de 200 a 600 px, cuyo punto medio corresponde a 400 px. De manera análoga, el segundo bin presenta un punto medio de 800 px, y así sucesivamente hasta el bin 18, cuyo punto medio es de 7200 px. Estos puntos medios fueron empleados como posiciones representativas de cada categoría de tamaño para el cálculo de la distancia de Wasserstein (EMD), considerando un ancho uniforme de 400 px para todos los bins.

1.3 Error absoluto medio (MAE)

El MAE es una métrica estándar que se puede usar para evaluar el desempeño del método de segmentación. La Ecuación 11 muestra el error absoluto medio (MAE) (Hodson, 2022). El MAE no mide el error en el diámetro de una burbuja individual, sino la discrepancia promedio entre el fr y fm en los diferentes *bins* del histograma.

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|fr_i - fm_i| \tag{11}$$

Donde:

- n = número de *bins* o contenedores (18 *bins*, no tiene unidad física);
- fr_i = frecuencia real o conteo de las burbujas en el bin i;
- fm_i = frecuencia medida en el bin i;

Se usó la Ecuación 11 para encontrar la discrepancia promedio MAE_A entre el f_r y el fm_A , y la discrepancia promedio MAE_B entre el f_r y el fm_B. Al evaluar los histogramas, el método de segmentación B implementado en esta investigación exhibió un desempeño superior frente al método de segmentación A implementado por Wright (1999) al registrar un error absoluto medio (MAE_B) de 3.61 burbujas en comparación con las 4.89 burbujas obtenidas por el método de segmentación A. Esto representó una reducción relativa del error del 26.14 % (calculada como [4.89 - 3.61] / 4.89 × 100), lo que consolida al método de segmentación B de esta investigación como la medición de mayor exactitud.

La Tabla 15 muestra los 18 bins con sus respectivos puntos medios en píxeles, el f_r, el fm_A, el fm_B y el $|fr - f_A|$ donde la sumatoria de estas es la discrepancia promedio o el error absoluto medio MAE_A, y el $|fr - f_B|$ donde la sumatoria de estas es el error absoluto medio MAE_B. Al evaluar los histogramas, el método de

segmentación B implementado en esta investigación exhibió un desempeño superior frente al método de segmentación A implementado por Wright (1999) al registrar un error absoluto medio (MAE_B) de 3.61 burbujas en comparación con las 4.89 burbujas obtenidas por el método de segmentación A. Esto representó una reducción relativa del error del 26.14 % (calculada como [4.89 - 3.61] / 4.89 × 100), lo que consolida al método de segmentación B de esta investigación como la medición de mayor exactitud.

Tabla 15

Comparación del error absoluto medio (MAE)

N.° *bin*	Eje x: Área (px)	fr	f_A	f_B	$\lvert fr - f_A \rvert$	$\lvert fr - f_B \rvert$
1	400	82	28	56	54	26
2	800	22	29	30	7	8
3	1200	12	19	15	7	3
4	1600	13	15	9	2	4
5	2000	1	3	5	2	4
6	2400	4	10	6	6	2
7	2800	3	5	5	2	2
8	3200	2	3	1	1	1
9	3600	3	1	2	2	1
10	4000	2	2	5	0	3
11	4410	2	2	6	0	4
12	4800	0	1	4	1	4
13	5200	0	1	0	1	0
14	5600	1	2	2	1	1
15	6000	0	0	0	0	0
16	6400	3	1	1	2	2
17	6800	1	1	1	0	0
18	7200	1	1	1	0	0
		Error absoluto medio (MAE)			**MAE_A**	**MAE_B**
		Unidad: burbujas			4.89	3.61

Nota. El método de segmentación B fue más exacto porque redujo el MAE_B a 3.61 burbujas. Asimismo, logró una reducción del 26.14 % del MAE_A.

En la Tabla 15 se aprecia que el método A presenta una discrepancia de 54 burbujas en el primer *bin*, al estimar 28 burbujas cuando el valor de referencia fue 82. No obstante, debido a que el MAE calcula el promedio de las discrepancias absolutas entre todos los bins, este error de gran magnitud se atenúa al combinarse con los errores de las demás categorías. Con el propósito de evaluar con mayor sensibilidad la influencia de estas discrepancias significativas, se empleó el error cuadrático medio (MSE), métrica que amplifica la contribución de los errores más severos al elevar cada diferencia al cuadrado antes de calcular su promedio.

1.4 Error cuadrático medio (MSE) y raíz del error cuadrático medio (RMSE)

De acuerdo con Hodson (2022), el RMSE corresponde a la raíz cuadrada del MSE, $RMSE = \sqrt{MSE}$, por lo que ambas medidas se encuentran directamente relacionadas como se muestra en la Ecuación 12 y la Ecuación 13. El MSE se presenta en unidades al cuadrado y el RMSE mantiene la unidad de la variable.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(fr_i - fm_i)^2 \tag{12}$$

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(fr_i - fm_i)^2} \tag{13}$$

Donde:

- n = número total de *bins* (18 *bins*, no tiene unidad física)
- fr_i = frecuencia de referencia o conteo real de burbujas en el bin i;
- fm_i = frecuencia medida en el *bin* i;

El MSE penaliza con mayor intensidad los errores grandes, pero no puede considerarse una métrica de distancia válida debido a que no satisface la desigualdad triangular y pierde la unidad de medida; por el contrario, la RMSE conserva las unidades originales, cumple las propiedades de una métrica de distancia y resalta las discrepancias de mayor magnitud, mientras que el MAE otorga el mismo peso a todos los errores. (Chai y Draxler, 2014)

La elección de una métrica de error debe fundamentarse en el comportamiento de los errores observados, ya que ninguna métrica puede considerarse superior en términos absolutos. Bajo esta perspectiva, el MAE presenta la ventaja de una mayor robustez frente a valores atípicos extremos, contribuyendo a la obtención de inferencias más estables y confiables. En consecuencia, la RMSE es más apropiada cuando los errores siguen una distribución normal (gaussianos), mientras que el MAE adquiere mayor relevancia cuando existe una mayor incidencia de errores extremos (laplacianos). (Hodson, 2022)

En la Tabla 16 vemos que el MSE_A es igual a 170.78 burbujas^2, mientras que el MSE_B se reduce a 46.50 burbujas^2. Esta diferencia representa que el MSE_B logró una reducción sustancial del 72.77 % del MSE_A (calculada como [170.78 - 46.50] / 170.78 × 100). Esta diferencia evidencia el efecto de la penalización cuadrática característica del MSE. Aunque la métrica incorpora el error de todos los *bins*, su valor se ve influenciado principalmente por aquellos que presentan las discrepancias más elevadas. Por ejemplo, en el primer *bin*, vemos que el cuadrado de la diferencia entre fr y el f_A del primer *bin* es 2916 burbujas^2, contribuyendo significativamente al incremento del MSE_A. En consecuencia, el MSE permite identificar con claridad las discrepancias de mayor magnitud, lo que lo convierte en una métrica altamente sensible a los errores grandes. Sin embargo, esta misma sensibilidad puede dificultar su interpretación, ya que una única observación con un error considerable puede influir de manera desproporcionada en el valor promedio. Adicionalmente, el MSE pierde la unidad

original de medida, ya que al elevar los errores al cuadrado sus resultados se expresan en burbujas^2.

Tabla 16

Comparación del error cuadrático medio (MSE)

N.° *bin*	Eje X: Área (px)	fr	f_A	f_B	$(fr - f_A)^2$	$(fr - f_B)^2$
1	400	82	28	56	2916	676
2	800	22	29	30	49	64
3	1,200	12	19	15	49	9
4	1,600	13	15	9	4	16
5	2,000	1	3	5	4	16
6	2,400	4	10	6	36	4
7	2,800	3	5	5	4	4
8	3,200	2	3	1	1	1
9	3,600	3	1	2	4	1
10	4,000	2	2	5	0	9
11	4,400	2	2	6	0	16
12	4,800	0	1	4	1	16
13	5,200	0	1	0	1	0
14	5,600	1	2	2	1	1
15	6,000	0	0	0	0	0
16	6,400	3	1	1	4	4
17	6,800	1	1	1	0	0
18	7,200	1	1	1	0	0
Error cuadrático medio (MSE)					**MSE_A**	**MSE_B**
Unidad: burbujas^2					170.78	46.50

Nota. El método de segmentación B indica una mejor exactitud (MSE_B = 46.50) en comparación con el método A (MSE_A = 170.78). Asimismo, logró una reducción del 72.77 % del MSE_A.

Para conservar la capacidad de enfatizar los errores de gran magnitud y, al mismo tiempo, mantener la interpretabilidad de la métrica en la unidad original de la variable (burbujas), puede emplearse la raíz del error cuadrático medio (RMSE). La Tabla 17 muestra que el método B alcanzó un $RMSE_B$ de 6.82 burbujas frente a las 13.07 burbujas obtenidas por el método A; por lo tanto, el método B logró reducir la magnitud del error $RMSE_B$ en un 47.82 % respecto al método A. Este resultado indica que las estimaciones generadas por el método B se aproximaron mejor a los valores reales.

Tabla 17

Comparación de la raíz del error cuadrático medio (RMSE)

Raíz del error cuadrático medio (RMSE)	$\mathbf{RMSE_A}$	$\mathbf{RMSE_B}$
Unidad: burbujas	13.07	6.82

Nota. El método de segmentación B presentó un menor $RMSE_B$ (6.82 burbujas), con una reducción del 47.82 % del $RMSE_A$ (13.07 burbujas).

La ventaja del RMSE es que la raíz cuadrada devuelve la métrica a la unidad original y tiene sensibilidad a los errores críticos, como se puede ver en el primer *bin*. Una desventaja del RMSE es que penaliza los errores grandes y es por eso que el $RMSE_A$ (13.068 burbujas) es mucho mayor que el MAE_A (4.88 burbujas).

1.5 La distancia de Wasserstein (EMD)

Según Ni *et al*. (2009), la distancia de Wasserstein corresponde a una medida de transporte óptimo utilizada para cuantificar la disimilitud entre histogramas. De igual forma, señalan que la distancia de Wasserstein constituye una métrica formal que satisface la desigualdad triangular. El *Earth Mover's Distance* (EMD) es equivalente a la distancia de Wasserstein cuando las distribuciones comparadas poseen la misma masa total o han sido normalizadas como funciones de densidad de probabilidad discretas. En estas condiciones, ambas métricas representan el costo mínimo o trabajo necesario para transformar una distribución en otra, por lo que producen el mismo valor de distancia. La formulación matemática empleada para su cálculo se presenta en la Ecuación 14, donde F_P es la función de distribución acumulada del histograma real P y F_Q es la función de distribución acumulada del histograma medido Q.

$$\mathrm{W}(P,Q) = \int_R \left|F_P(x) - F_Q(x)\right| dx \tag{14}$$

Cuando se dispone de histogramas discretos como es nuestro caso, la integral de la Ecuación 14 se aproxima mediante una sumatoria de Riemann como se muestra en la Ecuación 15. En consecuencia, esta ecuación fue utilizada para obtener la distancia de Wasserstein, la cual permitió comparar los histogramas normalizados de la distribución del tamaño de burbujas (BSD). La distancia de Wasserstein representa el trabajo mínimo requerido para transformar la silueta del histograma medido en el histograma real, o también se puede decir que representa el costo de mover masa de la distribución de las burbujas medidas para igualarla a la distribución de las burbujas reales. Bajo este fundamento, una menor distancia de Wasserstein indica una menor discrepancia entre las distribuciones comparadas y, por consiguiente, una mayor similitud respecto a la distribución de referencia.

$$\mathrm{W}(P,Q) \approx \sum_{i=1}^{n} \left| F_P(x_i) - F_Q(x_i) \right| \times \Delta x_i \tag{15}$$

Donde:

- $\mathrm{W}(P,Q)$ = distancia de Wasserstein entre el BSD P y el BSD Q (en píxeles, px);
- F_P = función de distribución acumulada (CDF) de las burbujas reales;
- F_Q = función de distribución acumulada (CDF) de las burbujas medidas;
- Δx = ancho físico del *bin*, (400 px);
- n = número total de contenedores (18 *bins*).

La Ecuación 16 muestra la función de distribución acumulada (CDF) de las burbujas reales $F_P(x_i)$, y la Ecuación 17 muestra la función de distribución acumulada (CDF) de las burbujas medidas $F_Q(x_i)$, donde sus valores van de 0 a 1.

$$F_P(x_i) = \frac{\sum_{k=1}^{i} fr_k}{\sum_{j=1}^{n} fr_j} \tag{16}$$

$$F_Q(x_i) = \frac{\sum_{k=1}^{i} fm_k}{\sum_{j=1}^{n} fm_j} \tag{17}$$

Donde:

- i = *bin* actual
- n = número total de contenedores (18 *bins*).
- fr_j = frecuencia de referencia o conteo real de burbujas en el bin j;
- fr_k = frecuencia de referencia o conteo real de burbujas en el bin k;
- fm_j = frecuencia medida o conteo real de burbujas en el bin j;
- fm_k = frecuencia medida o conteo real de burbujas en el bin k.

La distancia de Wasserstein se calculó en Microsoft Excel (versión 2207) mediante la implementación de las Ecuaciones 16, 17 y 18. Posteriormente, los

resultados fueron validados utilizando la función `wasserstein_distance` de la biblioteca SciPy en Python (versión 3.12.7), la cual es ampliamente utilizada para el cálculo de esta métrica en distribuciones unidimensionales. La coincidencia de los resultados en ambos entornos confirmó la correcta implementación del procedimiento de cálculo, lo que garantiza la reproducibilidad y confiabilidad de los valores reportados en la Tabla 18. La distancia de Wasserstein representa el costo mínimo de redistribución necesario para transformar una distribución medida en la distribución real, considerando tanto la cantidad de burbujas que deben desplazarse como la distancia recorrida entre bins. El método de segmentación B presentó una distancia de Wasserstein ($W_{(P,Q_B)}$= 364.92 px) inferior a la del método A ($W_{(P,Q_A)}$=427.33 px), lo que representa una reducción del error del 14.60 % a favor del método B. Dado que el ancho de cada *bin* es de 400 px, el costo de redistribución del método B equivale, aproximadamente, al desplazamiento total de una población de burbujas a lo largo de 0.91 *bins* (364.92 / 400), mientras que el costo de distribución $W_{(P,Q_A)}$ es equivalente al desplazamiento total de una población de burbujas a lo largo de 1.07 bins (427.33/400). Esto indica que el método B requiere una menor reorganización de las burbujas entre intervalos de tamaño para reproducir la distribución original, evidenciando una mayor similitud global con la distribución real.

Tabla 18

Comparación de la distancia de Wasserstein obtenida

Bin i	Eje x (píxeles)	fr	f_A	f_B	F_P $real$	F_Q $medida_A$	F_Q $medida_B$	$\lvert F_P - F_{Q_A} \rvert$ $\times 400$	$\lvert F_P - F_{Q_B} \rvert$ $\times 400$
1	400	82	28	56	0.5395	0.2258	0.3758	125.47	65.45
2	800	22	29	30	0.6842	0.4597	0.5772	89.81	42.81
3	1,200	12	19	15	0.7632	0.6129	0.6779	60.10	34.12
4	1,600	13	15	9	0.8487	0.7339	0.7383	45.93	44.17
5	2,000	1	3	5	0.8553	0.7581	0.7718	38.88	33.38
6	2,400	4	10	6	0.8816	0.8387	0.8121	17.15	27.80
7	2,800	3	5	5	0.9013	0.8790	0.8456	8.91	22.27
8	3,200	2	3	1	0.9145	0.9032	0.8523	4.50	24.85
9	3,600	3	1	2	0.9342	0.9113	0.8658	9.17	27.38
10	4,000	2	2	5	0.9474	0.9274	0.8993	7.98	19.22
11	4,400	2	2	6	0.9605	0.9435	0.9396	6.79	8.37
12	4,800	0	1	4	0.9605	0.9516	0.9664	3.57	2.37
13	5,200	0	1	0	0.9605	0.9597	0.9664	0.34	2.37
14	5,600	1	2	2	0.9671	0.9758	0.9799	3.48	5.10
15	6,000	0	0	0	0.9671	0.9758	0.9799	3.48	5.10
16	6,400	3	1	1	0.9868	0.9839	0.9866	1.19	0.11
17	6,800	1	1	1	0.9934	0.9919	0.9933	0.59	0.05
18	7,200	1	1	1	1.0000	1.0000	1.0000	0.00	0.00
Conteo	total	152	124	149					
			Distancia de Wasserstein Unidad: píxeles					$W_{(P,Q_A)}$ 427.33	$W_{(P,Q_B)}$ 364.92

Nota. El método de segmentación B presenta mayor similitud con la distribución real al requerir un menor trabajo de transporte ($W_{(P,Q_B)}$ = 364.92), lo que representa una reducción del 14.60 % en la distancia de Wasserstein respecto al método A.

1.6 Comparación de las métricas de error para los BSD

El análisis comparativo de las métricas evaluadas (MAE, MSE, RMSE y distancia de Wasserstein) confirmó que el método de segmentación B de esta investigación

exhibió la mayor exactitud, como se muestra en la Tabla 19. Este comportamiento se atribuye a que registró las menores magnitudes de error y la mayor concordancia con la BSD real. Por consiguiente, se determina que es el método que representa con mayor fidelidad la distribución real del tamaño de las burbujas**.**

Tabla 19

Comparación de las métricas de evaluación de los métodos A y B

Métrica	Método A	Método B	Reducción del error relativo (%)	Unidad
MAE	4.89	3.61	26.14	burbujas
MSE	170.78	46.50	72.77	burbujas²
RMSE	13.07	6.82	47.82	burbujas
$W_{(P,Q)}$	427.33	364.92	14.61	píxeles

Nota. El método de segmentación B desarrollado en esta investigación presentó un desempeño superior al registrar los menores valores de error en todas las métricas en comparación con el método de segmentación A propuesto por Wright (1999).

Para evaluar las diferencias entre las distribuciones del tamaño de las burbujas (BSD) se emplearon las métricas MAE, MSE, RMSE y la distancia de Wasserstein. Las tres primeras realizan una comparación directa entre los valores reales y los medidos en cada bin, pero difieren en la forma en que cuantifican el error. El MAE mide el error absoluto promedio y ofrece una interpretación sencilla de las discrepancias; el MSE incrementa la penalización de los errores grandes al elevarlos al cuadrado; mientras que el RMSE resalta los errores grandes, pero expresa el resultado en la misma unidad, la cual es el número de burbujas. Por su parte, la distancia de Wasserstein evalúa la diferencia entre distribuciones considerando la distancia entre *bins*, por lo que es capaz de capturar cambios en la estructura global de la BSD que pueden no reflejarse completamente en las métricas de error convencionales. Ninguna métrica puede considerarse universalmente superior a las demás, ya que cada una enfatiza diferentes

características del error y cada métrica se usa según la aplicación. Por tanto, la elección de una métrica depende del objetivo del análisis o de la aplicación: describir la discrepancia promedio de manera más estable mediante el MAE, o enfatizar diferencias importantes entre las predicciones y los datos reales mediante el RMSE, o ver la disimilitud entre histogramas de distribución en píxeles mediante la distancia de Wasserstein $W_{(P,Q)}$. Asimismo, tanto el RMSE como el MAE conservan la unidad original de número de burbujas, facilitando la interpretación de los resultados.

1.7 Discusión de la evaluación de los BSD mediante métricas de error

En esta investigación se encontró que el histograma real basado en la segmentación manual de Wright (1999), midió un total de 152 burbujas, de las cuales 82 burbujas se encuentran en el primer bin lo cual representa el 53.95 % de la población.

El método de segmentación B midió 56 burbujas y el método A midió muchísimo menos (28 burbujas) y eso causó que el método A tenga un mayor MAE_A (4.89 burbujas) que el método B (MAE_B= 3.61 burbujas), esto quiere decir que el BSD basado en el algoritmo de esta investigación (método B) es más exacto que el BSD basado en el método A, donde el MAE_B logró una reducción del 26.14 % del MAE_A. Luego el $RMSE_A$ es igual a 13.07 burbujas, lo cual es mayor que el MAE_A porque resalta los errores críticos del primer *bin*. Por otro lado, el MSE_A es 170.78 burbujas^2, lo cual es mucho mayor que el MSE_B (46.50 burbujas^2) debido a la penalización severa del primer *bin,* pero no se puede interpretar porque la unidad está elevada al cuadrado. Bajo esta perspectiva o aplicación, se puede inferir que el MAE presenta la ventaja de una mayor robustez frente a valores atípicos extremos de burbujas pequeñas en los métodos de segmentación A y B del primer *bin*, haciéndolo más estable, confiable y menos sensible que el MSE y RMSE. Sin embargo, si la aplicación es identificar errores significativos, entonces se recomienda el MSE. Y si la aplicación es enfatizar diferencias importantes y evaluar la exactitud global manteniendo la unidad, se recomienda el RMSE.

Finalmente, si el objetivo es evaluar la similitud global entre el BSD de referencia y los BSD generados por los métodos A y B, se recomienda utilizar la distancia de Wasserstein, ya que considera tanto la magnitud de las diferencias como su desplazamiento entre categorías de tamaño. Sin embargo, su interpretación física es menos intuitiva y más compleja que la de métricas como el MAE, el MSE o el RMSE. En este caso el $W_{(P,Q_A)}$ es igual a 427.33 px lo cual es mayor al $W_{(P,Q_B)}$ (364.92 px), lo que significa que el $W_{(P,Q_B)}$ requiere un menor trabajo de transporte que el $W_{(P,Q_A)}$, es decir que requiere menor reorganización de las burbujas entre intervalos de tamaño para reproducir la distribución original, evidenciando una mayor similitud global con el BSD real. En la Tabla 20 se presenta la comparación de las métricas en función a las ventajas, desventajas y aplicaciones. En consecuencia, estas métricas aportan información complementaria sobre diferentes aspectos del BSD y deben seleccionarse de acuerdo con el propósito específico del análisis.

Tabla 20

Comparación de las métricas de evaluación

Métrica	Ventajas	Desventajas	Aplicación
MAE	Fácil de interpretar y mantiene la unidad original.	No penaliza fuertemente los errores grandes.	Para evaluar la discrepancia promedio de manera más estable.
MSE	Penaliza con mayor intensidad los errores grandes.	Se expresa en burbujas², dificultando su interpretación física.	Para identificar errores significativos.
RMSE	Mantiene la unidad original y facilita la comparación del desempeño.	Sensible a valores atípicos.	Para enfatizar diferencias importantes y para evaluar la exactitud global.
$W_{(P,Q)}$	Considera la forma global de la distribución y posee una interpretación física como trabajo o costo de transporte en píxeles.	Requiere una interpretación más compleja que las métricas clásicas de error.	Para evaluar la similitud entre histogramas reales y medidos.

Nota. Ninguna métrica puede considerarse universalmente superior a las demás, ya que su utilidad depende del objetivo de evaluación y de la aplicación específica.

La Tabla 21 resume las principales características de las métricas.

Tabla 21

Características de las métricas empleadas en la evaluación del BSD

Característica	MAE	MSE	RMSE	Wasserstein
Métrica estándar en evaluación de modelos	✓	✕	✓	✓
Fácil interpretación física	✓	✕	✓	◐
Mantiene las unidades originales	✓	✕	✓	✓
Compara valores *bin* a *bin*	✓	✓	✓	✕
Considera la distancia entre *bins*	✕	✕	✕	✓
Penaliza más los errores grandes	✕	✓	✓	✕
Evalúa la similitud global entre histogramas	✕	✕	✕	✓

Nota. ✓ indica que cumple; ✕ que no cumple; y ◐ indica cumplimiento parcial.

2. Resultados de la clasificación de los BSD por tamaño de burbujas

Para ajustar y suavizar las barras discretas de la distribución del tamaño de las burbujas (BSD), se empleó una función de suavizado *kernel*, representada mediante una curva continua de color rojo. La clasificación de los BSD se presenta en la Figura 66. Con base en el tamaño predominante de las burbujas, los BSD se clasificaron en las siguientes cuatro clases:

- Clase A: burbujas grandes (7-8 cm^2).
- Clase B: burbujas medianas (6-7 cm^2).
- Clase C: burbujas pequeñas (5-6 cm^2).
- Clase D: burbujas muy pequeñas (4-5 cm^2).

La clase A del BSD representa a las burbujas grandes, con áreas promedio comprendidas entre 7 y 8 cm^2. Como se observa en la Figura 66, la curva de suavizado Kernel correspondiente a la clase A presenta una menor concentración de frecuencias

en los intervalos de menor área. Por ejemplo, el primer *bin* registra aproximadamente 30 burbujas con un área cercana a 2.5 cm². Asimismo, la distribución se extiende hacia áreas mayores, lo que evidencia el predominio de burbujas grandes. Para el frame 30 del video *7s_CLASE_A.mp4*, se obtuvo un área promedio de 7.5 cm².

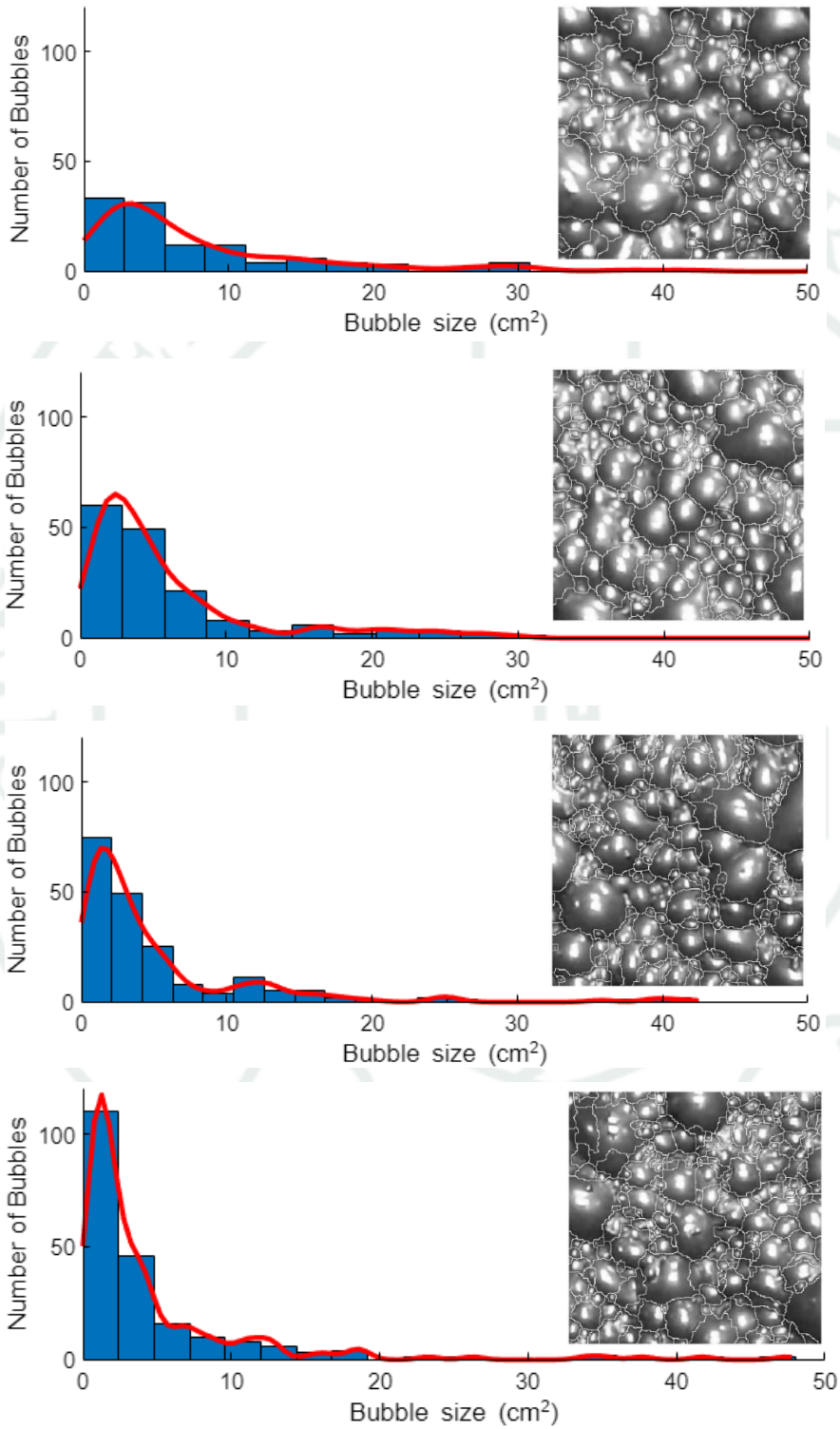
La clase B del BSD representa a las burbujas medianas, con áreas promedio comprendidas entre 6 y 7 cm². La curva de suavizado *kernel* presenta una mayor concentración de frecuencias en los intervalos de menor área en comparación con la clase A. Por ejemplo, el primer *bin* registra aproximadamente 60 burbujas con un área cercana a 2.5 cm². Para el frame 30 del video *5s_CLASE_B.mp4*, se obtuvo un área promedio de 6.4 cm².

La clase C del BSD representa a las burbujas pequeñas, con áreas promedio comprendidas entre 5 y 6 cm². La curva de suavizado *kernel* presenta una mayor concentración de frecuencias en los intervalos de menor área, registrándose aproximadamente 80 burbujas en el primer *bin*, con un área cercana a 2.5 cm². Para el frame 180 del video *6s_CLASE_D.mp4*, se obtuvo un área promedio de 5.4 cm².

La clase D del BSD representa a las burbujas muy pequeñas, con áreas promedio comprendidas entre 4 y 5 cm². La curva de suavizado Kernel presenta una alta concentración de frecuencias en los intervalos de menor área, registrándose aproximadamente 115 burbujas en el primer *bin*, con un área cercana a 2.5 cm². Para el frame 30 del *video 6s_CLASE_D.mp4*, se obtuvo un área promedio de 4.5 cm².

Figura 66

Clasificación de los BSD promedios en clases A, B, C y D



Clase A
Burbujas Grandes:
7 - 8 cm^2
Área (Frame 30) 7.5cm^2
Video 7s_CLASE_A.mp4

Clase B
Burbujas Medianas:
6 - 7 cm^2
Área (Frame 30) 6.4cm^2
Video 5s_CLASE_B.mp4

Clase C
Burbujas Pequeñas:
5 - 6 cm^2
Área (Frame 180) 5.4cm^2
Video 6s_CLASE_D.mp4

Clase D
Burbujas Muy Pequeñas:
4 - 5 cm^2
Área (Frame 30) 4.5cm2
Video 6s_CLASE_D.mp4

3. Discusión de la clasificación de los BSD

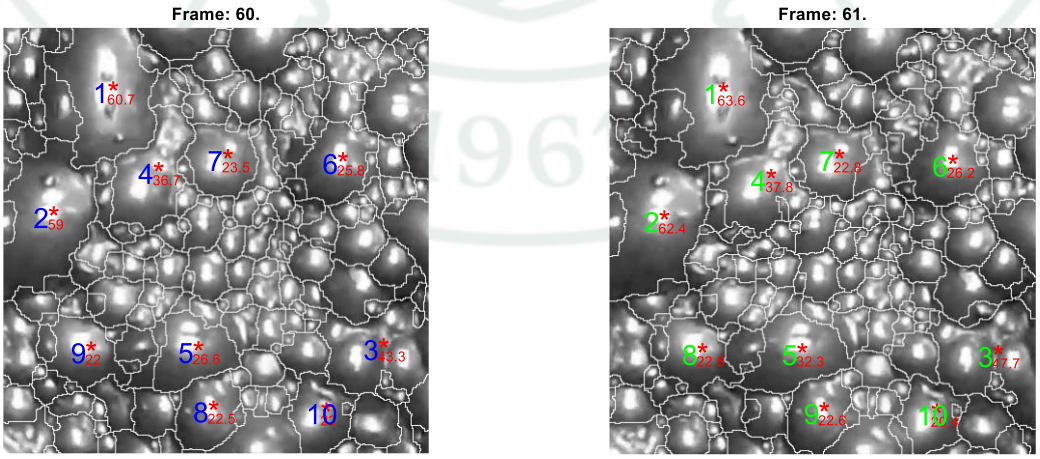
Los BSD fueron clasificados en las clases A, B, C y D, correspondientes a burbujas grandes, medianas, pequeñas y muy pequeñas, respectivamente. La clasificación de los BSD depende principalmente de la frecuencia de burbujas registrada en el primer *bin* (2.5 cm²). Las distribuciones de clase A presentan las menores frecuencias en este intervalo y corresponden a burbujas grandes, mientras que las distribuciones de clase D presentan las mayores frecuencias y corresponden a burbujas muy pequeñas. Esta variación se refleja en la altura de la curva *kernel* en el extremo izquierdo de la distribución, permitiendo identificar visualmente el tamaño predominante de las burbujas.

4. Resultados de la evaluación de la estabilidad, colapso y coalescencia

Para evaluar la estabilidad, se seleccionaron las 10 burbujas más grandes de cada frame. A continuación, se aplicó el algoritmo de estabilidad al video *6s_CLASE_D.mp4* para validar su desempeño. Como se observa en la Figura 67, los frames 60 y 61 registraron las mismas 10 burbujas mapeadas, lo que indica que conservaron su identidad en el segundo frame consecutivo.

Figura 67

La estabilidad de la espuma es 100 % a) Frame 60, b) Frame 61



a) b)

En consecuencia, el algoritmo clasificó las 10 burbujas como estables. Adicionalmente, se observó que ninguna de las burbujas analizadas presentó eventos de colapso o coalescencia en el segundo frame consecutivo. Aplicando la Ecuación 3, se obtuvo una estabilidad del 100 %, por lo que las 10 burbujas fueron clasificadas como estables.

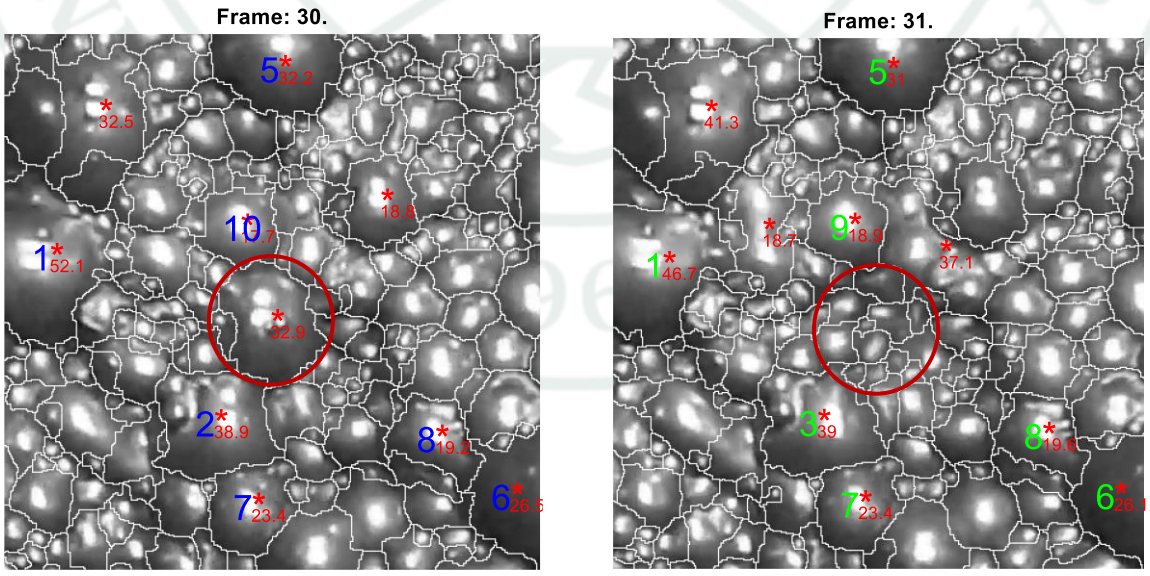
$$E = \frac{n}{N} \times 100 = \frac{10}{10} \times 100 = 100\%$$

En la Figura 68 se presenta una segunda medición de estabilidad. Los frames 30 y 31 mostraron que, de las 10 burbujas analizadas, 7 fueron clasificadas como estables, 1 colapsó y 2 experimentaron coalescencia al unirse con otras burbujas y formar una burbuja de mayor tamaño. Aunque el algoritmo detectó la burbuja resultante de la coalescencia, esta no fue considerada estable debido a la modificación de su identidad original. Aplicando la Ecuación 3, se obtuvo una estabilidad del 70 %.

$$E = \frac{7}{10} \times 100 = 70\%$$

Figura 68

Estabilidad de la espuma del 70 %: a) frame 30; b) frame 31



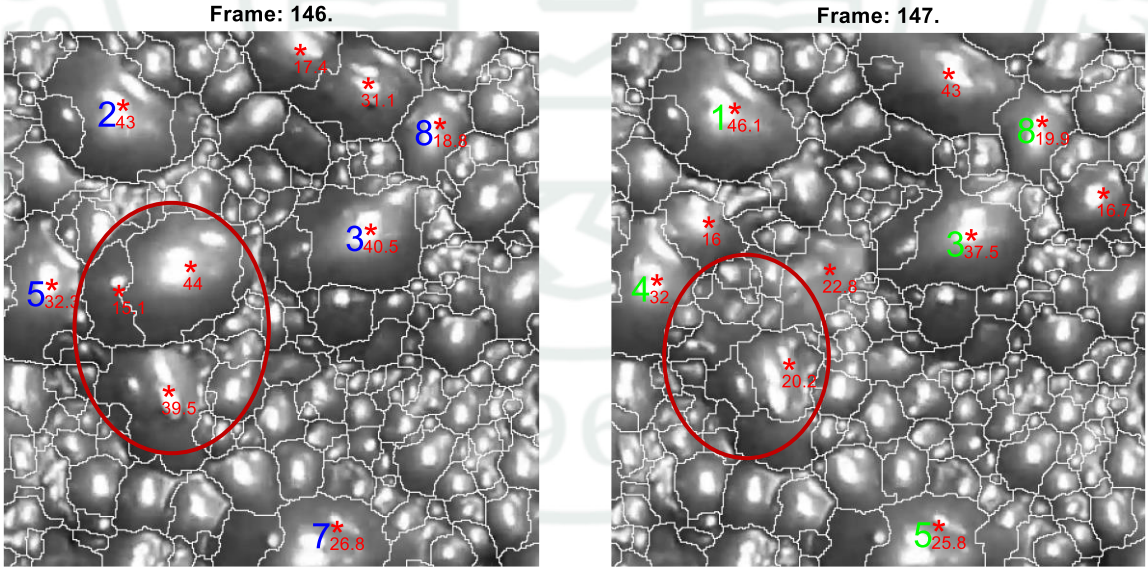
a) b)

En la Figura 69 se presenta la tercera y última medición de estabilidad. Los frames 146 y 147 muestran que, de las 10 burbujas grandes analizadas, 5 fueron clasificadas como estables, 3 burbujas contiguas colapsaron y 2 experimentaron coalescencia al unirse físicamente y formar una burbuja de mayor tamaño. Aunque el algoritmo detectó la burbuja resultante de la coalescencia, esta no fue considerada estable debido a la modificación de la identidad de las burbujas originales. Aplicando la Ecuación 3, se obtuvo una estabilidad del 50 %.

$$E = \frac{50}{10} \times 100 = 50\%$$

Figura 69

Estabilidad de la espuma del 50 %: a) frame 146; b) frame 147



a) b)

La Tabla 22 muestra que la estabilidad tuvo un comportamiento predominante en las burbujas evaluadas, representando el 73.33 % de las observaciones. Asimismo,

se registraron eventos de colapso y coalescencia, cada uno con una frecuencia del 13.33%, lo que evidencia la naturaleza dinámica de la espuma de flotación.

Tabla 22

Resultados de estabilidad, colapso y coalescencia de las burbujas

Medición	Estables	Colapsan	Coalescen
1	10 (100 %)	0 (0 %)	0 (0 %)
2	7 (70 %)	1 (10 %)	2 (20 %)
3	5 (50 %)	3 (30 %)	2 (20 %)
Total	22 (73.33 %)	4 (13.33 %)	4 (13.33 %)

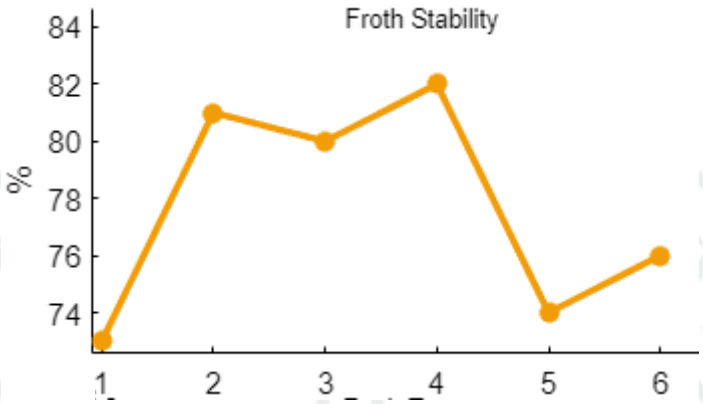
Nota. La estabilidad fue cuantificada mediante el algoritmo propuesto; en contraste, la coalescencia y el colapso fueron identificados mediante observación visual.

Después de obtener la estabilidad de la espuma para cada par de frames consecutivos, se calcularon 15 mediciones por segundo de video, considerando una velocidad de adquisición de 30 fps. Posteriormente, estas mediciones se promediaron para obtener un valor representativo de estabilidad por segundo. Finalmente, los valores resultantes se muestrearon en intervalos de 1 s para su visualización en el SCADA y el Dashboard IoT.

En la Figura 70, se muestra la evolución de la estabilidad de la espuma durante 6 s para el video *6s_CLASE_D.mp4*. Se observa que la estabilidad aumentó de 73 % a 81 % entre el primer y segundo intervalo de tiempo, disminuyó ligeramente a 80 % en el tercer segundo y alcanzó un valor máximo de 82 % en el cuarto segundo. Posteriormente, descendió a 74 % en el quinto segundo y finalizó en 76 % en el sexto segundo. En general, la estabilidad se mantuvo entre 73 % y 82 % durante el período analizado.

Figura 70

Medición de la estabilidad de la espuma durante 6 segundos



5. Discusión de los resultados de estabilidad de la espuma

Según Farrokhpay (2011), la coalescencia forma parte de la dinámica de la espuma y está relacionada con la estabilidad. Una espuma más estable presenta menos eventos de coalescencia y colapso, pero también puede favorecer la recuperación de material valioso y aumentar la recuperación de ganga por arrastre. Por ello, existe una estabilidad óptima para cada celda de flotación y sus condiciones de operación, tales como el espumante, el colector, el pH, la cantidad de partículas hidrofóbicas (mineral y ganga), el BSD, el caudal de aire y la densidad de la pulpa. Estas variables pueden influir significativamente en la estabilidad de la espuma y en el desempeño global de la flotación.

De acuerdo con la Tabla 22, el 73.33 % de las burbujas fueron clasificadas como estables, mientras que el colapso y la coalescencia representaron el 13.33 % cada uno. Estos resultados indican que la espuma evaluada presentó una estabilidad predominante, aunque continuó experimentando fenómenos dinámicos de inestabilidad. Asimismo, la presencia de eventos de colapso y coalescencia evidencia que la espuma no alcanzó una condición de estabilidad absoluta. Por ello, los

resultados obtenidos evidencian la coexistencia de fenómenos de estabilidad, colapso y coalescencia en la espuma analizada.

Adicionalmente, la Figura 70 muestra que la estabilidad de la espuma fluctuó entre 73 % y 82 % durante los 6 s analizados. Se observaron incrementos y disminuciones a lo largo del tiempo, alcanzando un valor máximo de 82 % y un valor mínimo de 73 %. Este comportamiento indica que la estabilidad de la espuma no permaneció constante durante el período evaluado. Asimismo, estas fluctuaciones evidencian que la estabilidad de la espuma es una variable dinámica que varía continuamente en función de la evolución de las burbujas.

5.1 Propuesta de umbrales de alarma para la estabilidad de la espuma

Con el objetivo de facilitar la interpretación de la estabilidad de la espuma, se propusieron umbrales para la generación de alarmas. La variable de estabilidad se representó mediante la sigla S. Para la clasificación de los niveles de estabilidad se definieron cuatro categorías: muy alto (SHH), alto (SH), bajo (SL) y muy bajo (SLL). Los umbrales propuestos se presentan en la Tabla 23.

Tabla 23

Clasificación de alarmas según el nivel de estabilidad de la espuma

Alarmas para la estabilidad	Rango
Estabilidad muy alta (SHH)	$80 < S \leq 100$
Estabilidad alta (SH)	$70 < S \leq 80$
Estabilidad baja (SL)	$60 < S \leq 70$
Estabilidad muy baja (SLL)	$50 < S \leq 60$

Nota. S representa la estabilidad de la espuma expresada en porcentaje.

6. Resultados de la velocidad de la espuma

Aplicando la Ecuación 9, se obtuvo una velocidad promedio total (Vm_T) de 22.67 cm/s en 1 segundo de video, como se muestra en la Tabla 24.

Tabla 24

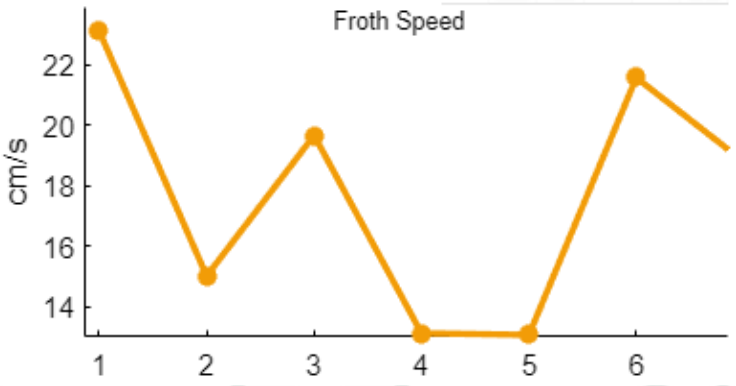
Velocidad promedio total Vm_T obtenida en 1 segundo de video

i	# Frame	$\bar{V}m_i$ cm/s
1	2	34.300
2	4	27.325
3	6	14.229
4	8	13.500
5	10	22.443
6	12	31.414
7	14	17.457
8	16	6.975
9	18	12.114
10	20	28.980
11	22	15.450
12	24	40.280
13	26	45.775
14	28	13.367
15	30	16.417
Velocidad promedio total en 1 segundo Vm_T		22.67

Finalmente, se implementó en el SCADA y en el Dashboard IoT la gráfica de tendencia para la velocidad promedio total Vm_T en cm/s para 7 segundos, como se muestra en la Figura 71.

Figura 71

Medición de la velocidad de la espuma durante 7 s del video

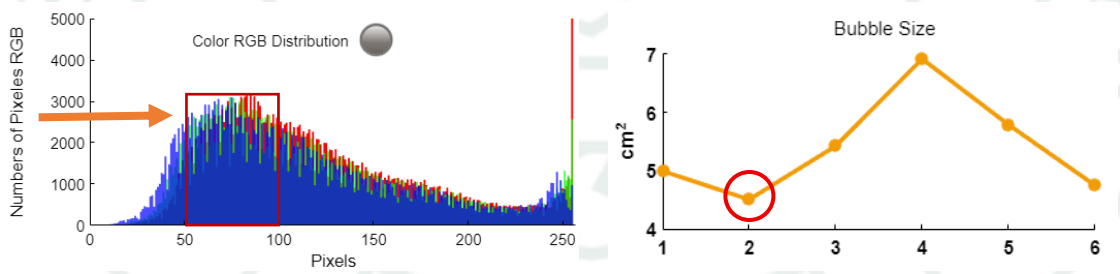


7. Resultados de la distribución del color de la espuma (FCD)

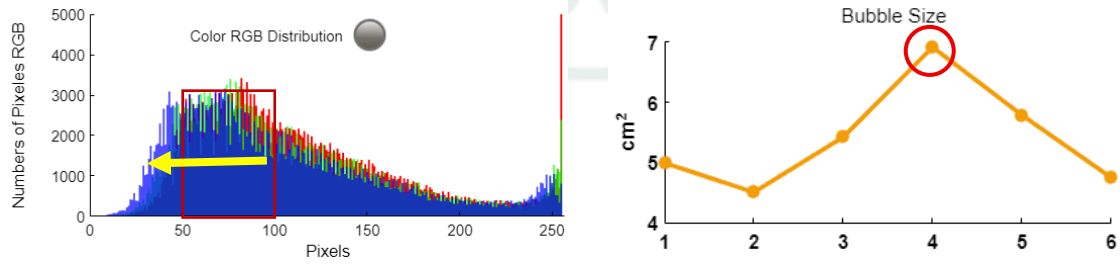
La distribución de los colores RGB de la espuma, Froth Color Distribution (FCD), se presenta en la Figura 72 a partir del video 6s_CLASE_D.mp4.

Figura 72

Influencia del tamaño de las burbujas en la distribución del color de la espuma (FCD)



a)



b)

Nota. La reducción del tamaño de las burbujas desplaza el FCD hacia la derecha (mayor intensidad) debido a una mayor reflexión de la luz en las burbujas pequeñas, mientras que el aumento del tamaño de las burbujas lo desplaza hacia la izquierda (menor intensidad).

El eje X representa los niveles de intensidad de color, con valores entre 0 y 255, mientras que el eje Y representa la frecuencia, expresada como el número de píxeles asociados a cada nivel de intensidad. La distribución muestra que la espuma está compuesta principalmente por píxeles de intensidad media, concentrados entre 50 y 120 niveles de color. Asimismo, el canal rojo presenta frecuencias ligeramente superiores respecto a los canales verde y azul en gran parte del intervalo analizado. Finalmente, se observa un pico pronunciado en la intensidad máxima (255), especialmente en el canal rojo, lo que evidencia la presencia de regiones de alta luminosidad asociadas a reflejos en la superficie de la espuma.

En la Figura 72 (a), correspondiente al intervalo 2, se observa el menor tamaño promedio de burbuja (4.5 cm^2). En este intervalo, el FCD presenta una mayor concentración de píxeles en intensidades elevadas, evidenciando un desplazamiento hacia la derecha del histograma. Este comportamiento puede atribuirse a que las burbujas más pequeñas generan una mayor reflexión de la luz sobre la superficie de la espuma, produciendo una apariencia más clara.

Por otro lado, en la Figura 72 (b), correspondiente al intervalo 4, se observa el mayor tamaño promedio de burbuja (6.9 cm^2). En este caso, el FCD presenta un desplazamiento hacia intensidades menores, es decir, hacia la izquierda del histograma, indicando una apariencia más oscura de la espuma.

Estos resultados sugieren una relación inversa entre el tamaño de las burbujas y la intensidad de color registrada. A medida que disminuye el tamaño de las burbujas, aumenta la reflexión de la luz sobre la superficie espumosa, incrementando la concentración de píxeles en valores de intensidad más altos y desplazando el FCD hacia la derecha del histograma. En contraste, las burbujas de mayor tamaño reflejan menos luz, favoreciendo una distribución asociada a intensidades más bajas.

8. Resultados de la energía para la caracterización de la textura de la espuma

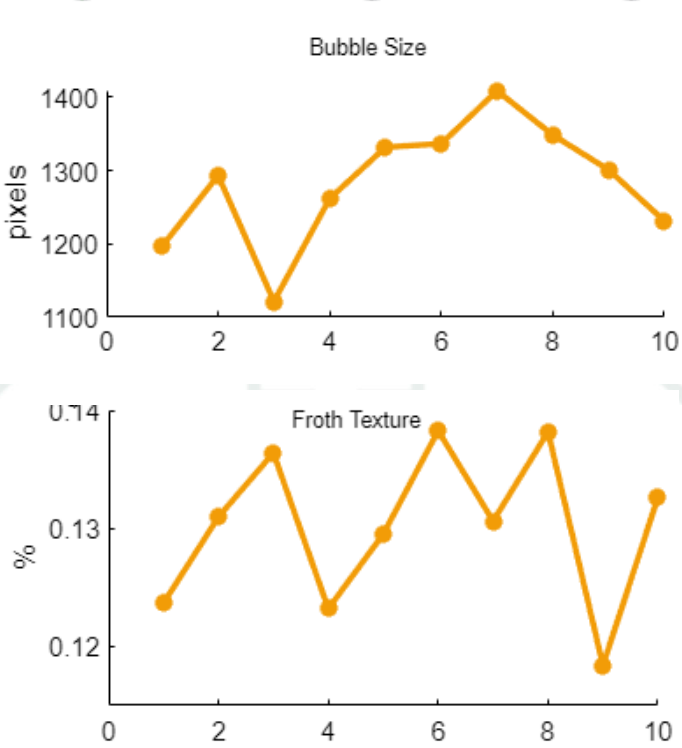
Cada 30 frames se obtuvo el valor de energía, cuyos valores se encuentran entre 0 y 1. Para analizar su comportamiento, se tomaron diez muestras en intervalos de 1 s del video.

La relación entre la energía de textura y el tamaño de las burbujas se presenta en la Figura 73, donde se observó una tendencia similar entre la evolución del tamaño de las burbujas y la energía de textura. La energía puede tomar valores entre 0 y 1, donde valores más altos indican una mayor uniformidad textural. En general, cuando el tamaño de las burbujas aumentó, la energía también tendió a incrementarse, alcanzando valores cercanos a 0.14 en las muestras 6 y 8, donde también se registraron áreas elevadas de burbuja.

Asimismo, cuando el tamaño de las burbujas disminuyó, la energía mostró una tendencia a reducirse, registrando valores próximos a 0.12 en las muestras 4 y 9. Sin embargo, esta relación no se cumplió en todos los casos; por ejemplo, en la muestra 7 se registró el mayor tamaño de burbuja, mientras que la energía disminuyó respecto a la muestra anterior. Estas discrepancias pueden atribuirse a la naturaleza dinámica de la espuma, donde fenómenos como la coalescencia, el colapso y los cambios en la distribución espacial de las burbujas modifican la textura observada en la imagen.

Figura 73

Relación entre la energía de textura y el tamaño de las burbujas



Nota. En general, el incremento del tamaño de las burbujas estuvo asociado con un aumento de la energía de textura, con algunas variaciones puntuales derivadas de la dinámica de la espuma.

9. Resultados de los tiempos computacionales de los algoritmos

La medición de los tiempos computacionales es importante porque nos permite determinar el tiempo de muestreo o intervalo que se mostrará en el SCADA y el Dashboard IoT. Cada muestra se envía al Dashboard IoT en un intervalo de tiempo. Para medir los algoritmos podemos usar la función *tic* y *toc* de MATLAB®. Con *tic* se registra el tiempo actual y con *toc* se calcula el tiempo transcurrido. Con *tic toc*, los resultados pueden ser variables para los códigos pequeños por debajo de 0.100 s debido a las sobrecargas que se produzcan en el sistema (sistema operativo, protocolos

de red, memoria, ancho de banda, etc.). Para medidas más precisas se puede usar *timeit*, pero implicaría juntar los códigos en funciones y eso aumentaría más el tiempo computacional. Con MATLAB Profiler se puede obtener un reporte en HTML que permite verificar el desempeño y los cuellos de botella que hay en el código.

Los códigos que usamos no son pequeños; por lo tanto, las funciones *tic toc* son suficientes para medir el tiempo computacional de los algoritmos de esta tesis. En la Tabla 25, vemos los tiempos que toman cada algoritmo. Para medir la velocidad de la espuma, se usaron dos frames. El tiempo computacional para el tamaño de las burbujas en el primer frame fue de 0.108 s y en el segundo frame fue 0.105 s. Luego de tener los centroides de los dos frames, se pudo medir la velocidad. El tiempo computacional de la velocidad y la estabilidad fue de 0.193 s. El tiempo total en dos frames para el tamaño, la velocidad y la estabilidad fue de 0.406 s, casi medio segundo.

Tabla 25

Tiempo computacional de los algoritmos para el tamaño, estabilidad y velocidad

Algoritmos	Tiempo computacional (s)
Tamaño de burbuja - frame 1	0.107667
Tamaño de burbuja - frame 2	0.105218
Estabilidad y velocidad - Por cada 2 frames	0.192764
Tiempo total	0.405649

Nota. Tiempo computacional total de 0.41 s por cada par de frames, medido mediante las funciones *tic y toc*, para el cálculo del tamaño de burbuja, la estabilidad y la velocidad.

Debido a que el video es de 30 fps, se pudieron medir hasta 15 pares de frames. Luego, se obtuvo el promedio de las medidas para obtener una muestra. Se multiplicó 0.406 s por 15 y nos dio 6.084735 s. En cada 15 pares de frames, se midió el color y la textura en 0.022 s. Luego, el tiempo que se tomó en escribir los datos a un canal ThingSpeak es de 0.512 s, casi medio segundo. Por lo tanto, el tiempo computacional que se tomó para medir las características de la espuma y enviar una muestra a un canal ThingSpeak fue de 6.619 s, lo cual es la suma de todos estos tiempos como se muestra en la Tabla 26. Si queremos enviar más muestras, solo tenemos que multiplicar por este factor. Para enviar cinco muestras a un canal ThingSpeak™, se multiplicó 6.619 s por 5 y nos dio 33.094 s. Todos estos tiempos se midieron código por código en MATLAB®.

Tabla 26

Cálculo del tiempo computacional código por código para 1 muestra y para 5 muestras

Concepto	Tiempo computacional (s)
Tiempo total para tamaño de burbuja, velocidad y estabilidad (15 pares de frames)	6.084735
Color y textura en 15 pares de frames	0.0224272
Escritura en un canal de ThingSpeak cada 15 pares de frames	0.511623
Tiempo total para 1 muestra	6.6187852
Tiempo total para 5 muestras	33.093926

Nota. Una muestra corresponde a 15 pares de frames. El tiempo computacional total fue de 6.62 s por muestra y de 33.09 s para cinco muestras. Los tiempos fueron medidos código por código en MATLAB® mediante las funciones *tic y toc*.

En la Tabla 27 se muestra el tiempo computacional medido para la ejecución completa del código en MATLAB®, desde el inicio hasta el final del procesamiento. El tiempo total fue de 6.26 s para una muestra y de 27.44 s para cinco muestras.

Tabla 27

Tiempo computacional de la ejecución completa del código para 1 y 5 muestras en MATLAB®

N.° de muestras en MATLAB®	**Tiempo (s)**
Tiempo total para 1 muestra	6.26
Tiempo total para 5 muestras	27.44

Nota. El tiempo computacional para la ejecución completa del código fue de 6.26 s para 1 muestra y de 27.44 s para 5 muestras, medidos en MATLAB® desde el inicio hasta el final del procesamiento.

En la Figura 74, se presenta el reporte del desempeño con MATLAB Profiler. Se observa un tiempo computacional de 36.67 s para el procesamiento de cinco muestras.

Figura 74

Análisis del tiempo computacional para cinco muestras mediante MATLAB Profiler

video_bubble_watershed_Project_SCADA_v3
Profile Summary

Generated 01-Jan-2026 17:22:50 using performance time.

Function Name	Calls	Total Time (s)	Self Time* (s)	Total Time Plot (dark band = self time)
video_bubble_watershed_Project_SCADA_v3	1	36.765	1.106	
imshow	75	10.323	0.983	
initSize	75	7.257	0.061	
movegui	75	7.012	0.011	
legacyMoveGUI	75	6.980	2.717	
watershed	150	6.663	4.700	
thingSpeakWrite	5	4.988	0.008	

Nota. El tiempo computacional total fue de 36.67 s para cinco muestras. Las funciones *imshow* y *watershed* registraron tiempos acumulados de 10.33 s y 6.66 s, respectivamente.

La instrucción *imshow* registró el mayor tiempo de ejecución (10.33 s) debido a que se utiliza para representar los frames en forma de video. Esta función fue ejecutada 15 veces por muestra y 75 veces para cinco muestras. Por otro lado, la función *watershed* fue ejecutada 300 veces por que se usa en cada frame que se mide las burbujas y el tiempo computacional es de 6.663 s para 5 muestras y por cada frame demora 0.022 s.

En la Tabla 28, se muestran los tiempos computacionales correspondientes a la ejecución del SCADA implementado con MATLAB App Designer. Se observa que los tiempos obtenidos son menores que los registrados durante la ejecución directa en MATLAB®. El tiempo computacional fue de 5.58 s para una muestra y de 30.52 s para cinco muestras.

Tabla 28

Tiempo computacional del SCADA implementado con MATLAB App Designer

N.° de muestras en App MATLAB Designer	Tiempo (s)
Tiempo total para 1 muestra	5.58
Tiempo total para 5 muestras	30.52

Nota. El tiempo computacional fue de 5.58 **s** para 1 muestra y de 30.52 **s** para 5 muestras durante la ejecución del SCADA implementado con MATLAB App Designer.

Por lo tanto, el intervalo de actualización fue de aproximadamente 5 s por muestra. Este comportamiento también se verificó en la Tabla 29, donde el tiempo se presenta en formato hh:mm:ss, y en la Figura 75 donde también se muestra el eje x del ThingSpeak Axis con los tiempos en formato hh:mm:ss y se ve que cada muestra se da en aproximadamente 5 s.

10. Discusión de los tiempos computacionales

Los resultados obtenidos muestran que los tiempos computacionales dependen principalmente del procesamiento de imágenes y de la frecuencia de actualización de las muestras. El cálculo del tamaño de burbuja, la estabilidad y la velocidad requirió 0.41 s por cada par de frames, mientras que una muestra completa presentó tiempos entre 5.58 s y 6.62 s.

11. Resultados del SCADA y del Dashboard IoT

En la Tabla 29 se presentan cinco mediciones obtenidas en el SCADA y el Dashboard IoT a partir del video 6s_CLASE_D.mp4. Los resultados muestran una coincidencia total entre los valores de la velocidad, estabilidad y textura de la espuma, así como del área de la burbuja, registrados en ambas plataformas, verificando el correcto funcionamiento del sistema de monitoreo desarrollado.

Tabla 29

Comparación de las características de la espuma obtenidas en el SCADA y el Dashboard IoT

N.°	Time	Bubble size (cm²)		Speed (cm/s)		Stability (%)		Texture (0-1)	
Sample	hh/mm/ss	SCADA	IoT	SCADA	IoT	SCADA	IoT	SCADA	IoT
1	13:11:30	4.99	4.99	16.30	16.30	73	73	0.14	0.14
2	13:11:36	4.50	4.50	20.29	20.29	81	81	0.13	0.13
3	13:11:41	5.42	5.42	17.04	17.04	80	80	0.13	0.13
4	13:11:47	6.91	6.91	10.26	10.26	82	82	0.15	0.15
5	13:11:52	5.78	5.78	16.45	16.45	74	74	0.15	0.15
Total Time for 5 Samples: 27.67 s									

Nota. Los valores registrados en el SCADA y el Dashboard IoT fueron idénticos para todas las muestras, confirmando la trazabilidad de los datos entre ambas plataformas

En la Figura 75 se muestran las tendencias del área de la burbuja (*bubble size*), la velocidad (*speed*), la estabilidad (*stability*) y la textura (*texture*) de la espuma obtenidas a partir del video *6s_CLASE_D.mp4*. Se observa que, entre la primera y segunda medición, el área de la burbuja disminuyó de 4.99 cm² a 4.50 cm², mientras que la velocidad aumentó de 16.30 cm/s a 20.29 cm/s.

De manera similar, entre la segunda y tercera medición, el área aumentó de 4.50 cm^2 a 5.42 cm^2 y la velocidad disminuyó de 20.29 cm/s a 17.04 cm/s. Asimismo, en la cuarta medición se registró el mayor valor de área (6.91 cm^2) y el menor valor de velocidad (10.26 cm/s).

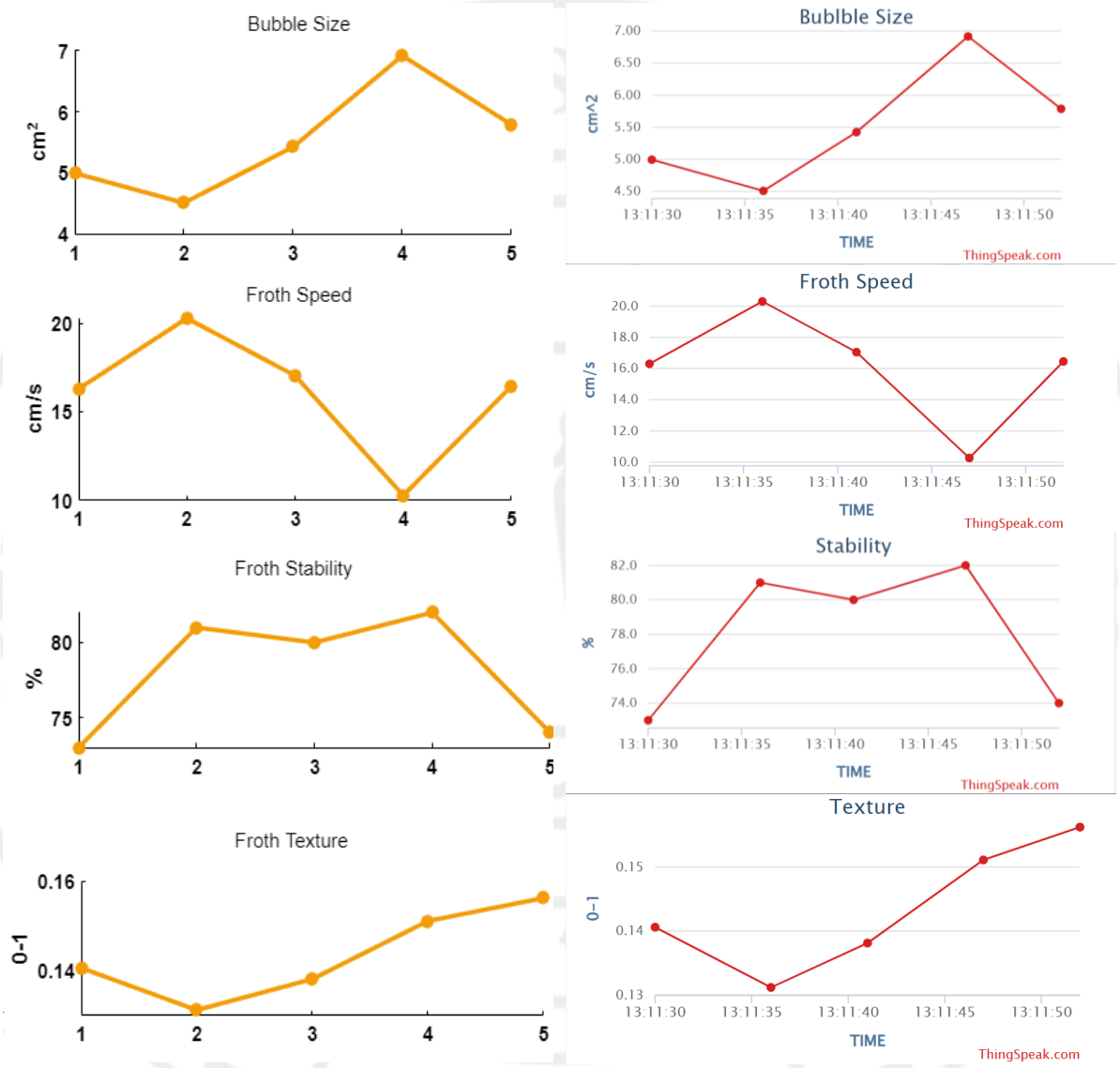
Estos resultados sugieren una tendencia inversa entre el tamaño de las burbujas y la velocidad de la espuma en las muestras analizadas. Estas relaciones no siempre se presentan de forma consistente debido a la dinámica y complejidad del proceso de flotación, ya que la espuma puede reducir su movimiento, detenerse temporalmente o desplazarse en diferentes direcciones, como hacia atrás o hacia los lados.

También se observa una tendencia inversa entre el área de las burbujas y la estabilidad de la espuma. Cuando el área disminuye, la estabilidad tiende a incrementarse, mientras que al aumentar el área la estabilidad tiende a reducirse. En la cuarta medición se registró el mayor valor de área de burbuja, acompañado de una disminución de la estabilidad.

Asimismo, se aprecia que alrededor de la tercera medición se alcanza un equilibrio entre las variables analizadas, con un área de 5.42 cm^2, una velocidad de 17.04 cm/s y una estabilidad cercana al 80 %. Estos resultados sugieren que tamaños de burbuja moderados podrían favorecer una mayor estabilidad de la espuma. La textura presentó una tendencia creciente conforme aumentó el área de las burbujas.

Figura 75

Tendencias de las características de la espuma con a) MATLAB App Designer *y b)* ThingSpeak™



a) b)

Nota. Las tendencias mostraron valores de estabilidad, velocidad, área y textura entre 73 % - 82 %, 10 - 20 cm/s, 4 - 7 cm² y 0.13 - 0.16, respectivamente. El área de burbuja presentó una relación inversa con la velocidad y la estabilidad, y directa con la textura.

12. Discusión de los resultados del SCADA y del Dashboard IoT

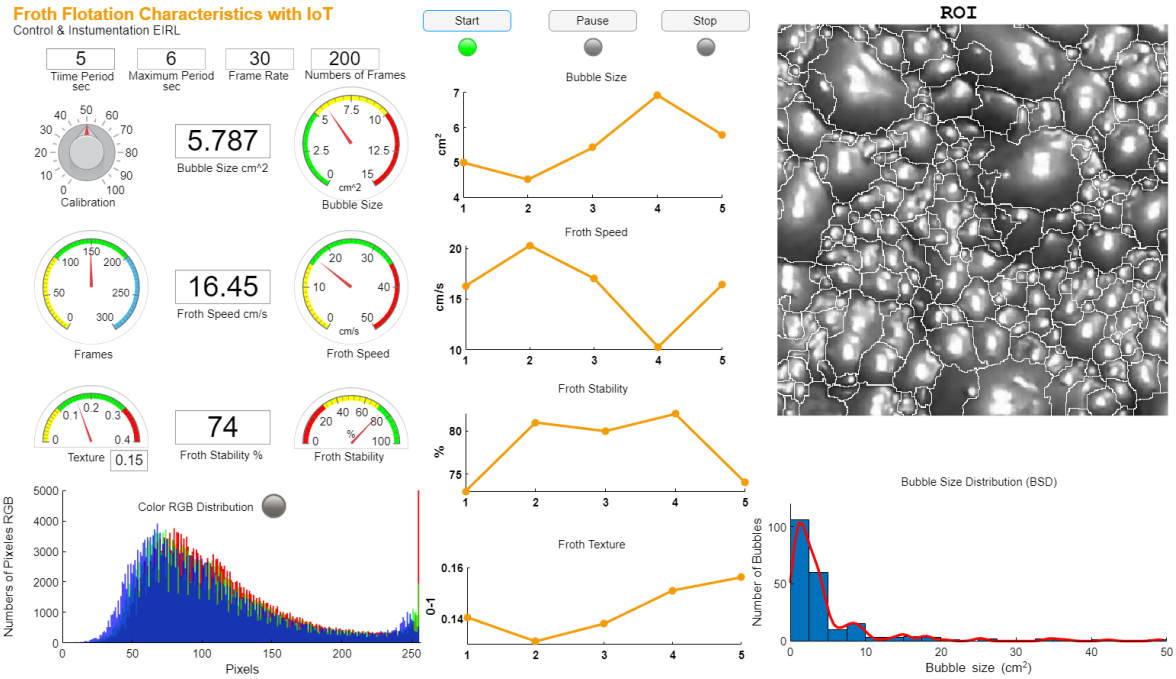
En conjunto, los resultados muestran que las imágenes de la espuma pueden segmentarse para medir sus características físicas, estáticas y dinámicas e integrarlas exitosamente en un SCADA y un Dashboard IoT donde se registraron exactamente los mismos valores de área de burbuja, velocidad, estabilidad y textura de la espuma, evidenciando la correcta adquisición, procesamiento, transmisión y visualización para su análisis y monitoreo bajo los principios de la Industria 4.0. Los resultados del análisis presentaron una tendencia inversa entre el tamaño de las burbujas y la velocidad de la espuma, resultado que coincide con lo reportado por Massinaei et al. (2019), quienes señalaron que un mayor contenido de sólidos favorece el incremento del tamaño de las burbujas y la disminución de la velocidad de la espuma. Una reducción del tamaño de las burbujas desplaza el FCD hacia la derecha, donde hay mayor intensidad de color debido a una mayor reflexión de la luz en las burbujas pequeñas. Se observó una tendencia directa entre el tamaño de las burbujas y la energía textural, resultado que parece relacionarse con Massinaei et al. (2019), quienes indicaron que la energía textural aumenta con el flujo de aire, es decir que las imágenes de la espuma parecen ser más uniformes en términos del BSD a mayores flujos de aire.

13. Evidencias en video de la investigación

En la Figura 76, vemos el SCADA con el nombre *Froth Flotation Characteristics with IoT*. Vemos que el video es de 30 fps, y el video contiene 200 frames y se puede medir las características hasta 6 segundos, pero se seleccionó para verlos por 5 segundos.

Figura 76

SCADA para monitorear las características de la espuma



En la Figura 77, vemos el código QR que direcciona al enlace de *internet* https://youtu.be/d7Gv2zamoTQ?si=dQWDS741w4A0Rjub para poder ver el video: *Froth Flotation Characteristics SCADA - Video 6s CLASE D*

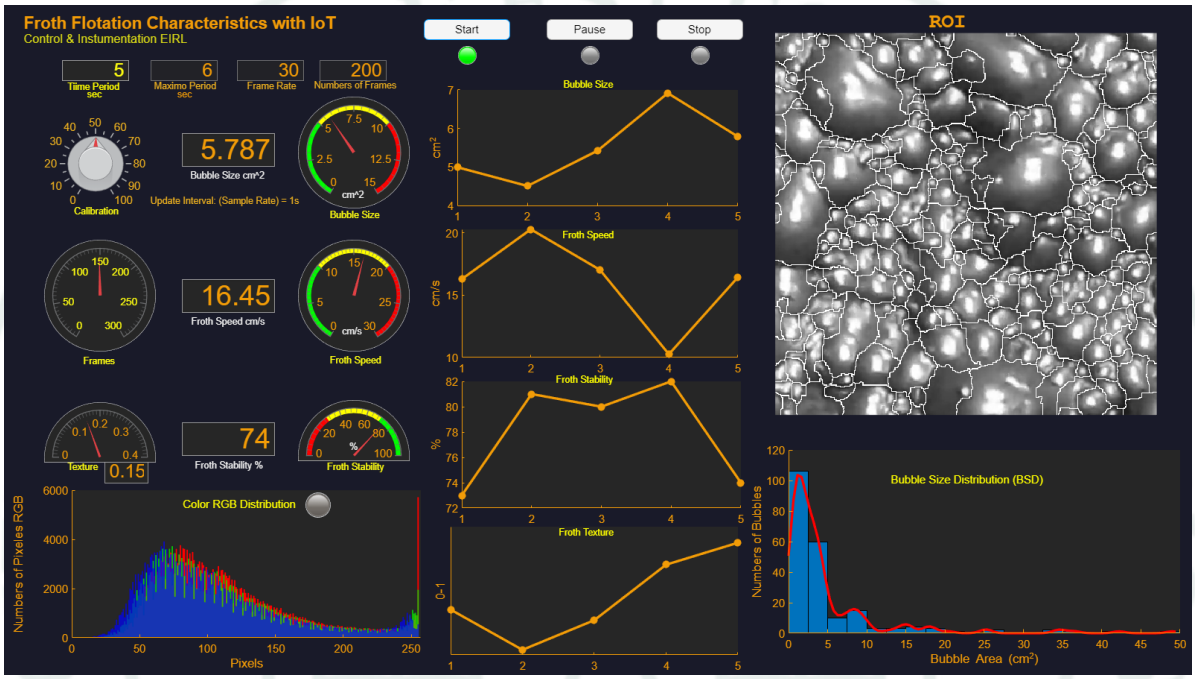
Figura 77

Código QR para el video Froth Flotation Characteristics SCADA - Video 6s CLASE D

En la Figura 78 vemos el mismo SCADA, pero en modo oscuro para operadores que desean descansar la vista.

Figura 78

SCADA en modo oscuro: características de la espuma



Enlace https://youtu.be/U4GAiXUZgHI?si=qnZbeeiaWglJ_MCU del QR para el video: *Froth Flotation Characteristics SCADA - Video 10s CLASE ABCD*, Figura 79

Figura 79

Código QR: video - Froth Flotation Characteristics SCADA Dark Mode

En la Figura 80, podemos ver el Dashboard IoT para analizar las características de la espuma desde ThingSpeak.

Figura 80

Dashboard IoT para analizar las características de la espuma



Se puede visualizar que los puntos se grafican en el SCADA, y se actualiza el Dashboard IoT cada 5 s con una latencia cerca de 0.5 s.

Este es el link https://youtu.be/HsOYgYjmYI8?si=Y5gFPif-J2SbmfF8 para el QR para poder ver el video: *IoT Dashboard & SCADA for Froth Flotation Characteristics*, como se muestra en la Figura 81.

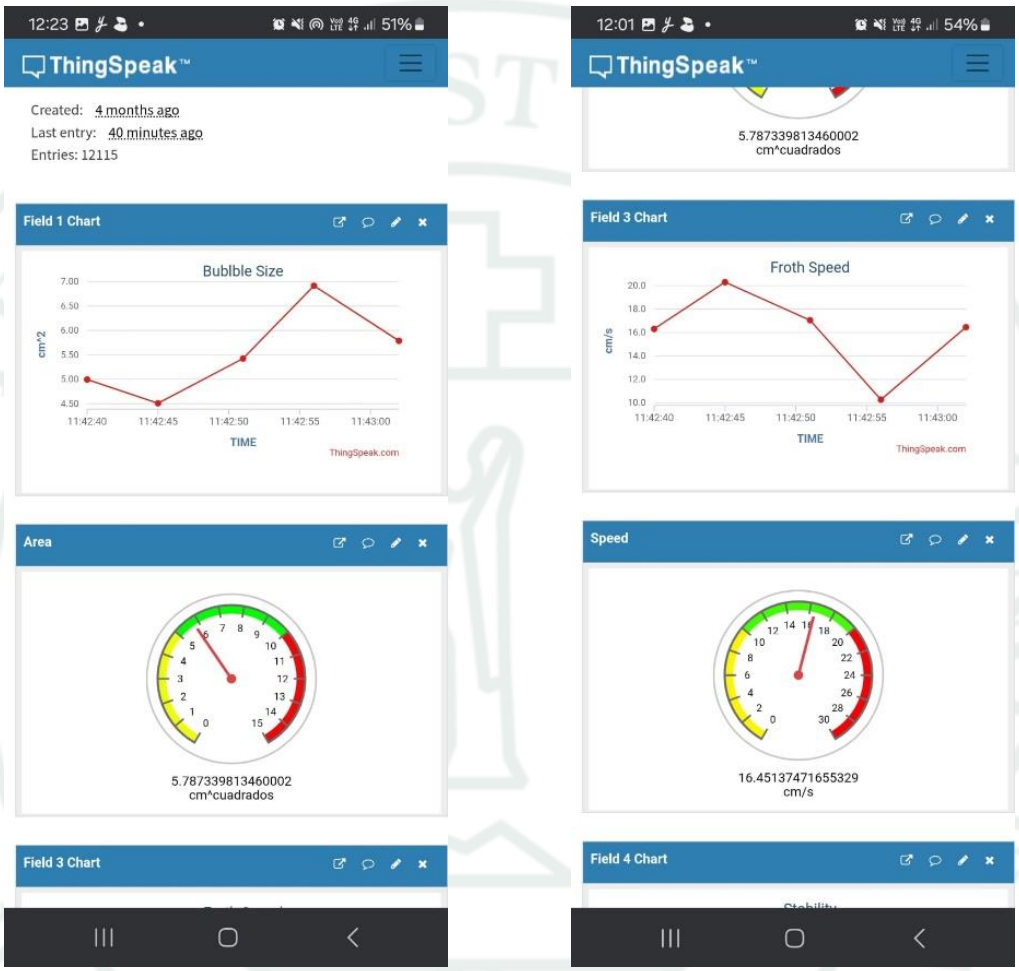
Figura 81

Código QR para el video IoT Dashboard & SCADA

A la izquierda de la Figura 82 se puede analizar en el Dashboard IoT las tendencias del tamaño de las burbujas junto con su indicador de tipo gauge. Y por la derecha, se puede analizar la velocidad de la espuma con su indicador de tipo gauge. Esto se pudo ver desde un smartphone Samsung A15 con conexión a internet.

Figura 82

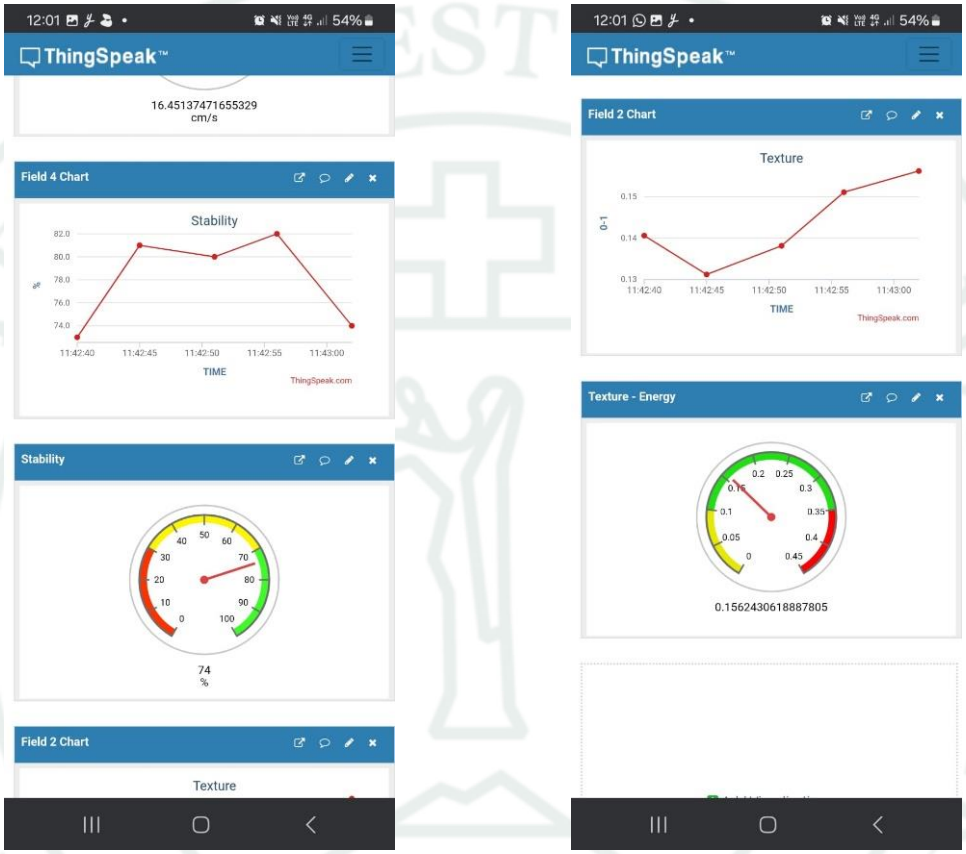
Dashboard IoT desde un smartphone Samsung A15 para ver el tamaño de las burbujas y la velocidad de la espuma



A la izquierda de la Figura 83 se puede analizar en el Dashboard IoT las tendencias de la estabilidad de la espuma junto con su indicador de tipo gauge. Y por la derecha, se puede analizar la textura de la espuma con su indicador de tipo gauge. Esto también se pudo ver desde un smartphone Samsung A15 con conexión a internet.

Figura 83

Dashboard IoT desde un smartphone Samsung A15 para ver la estabilidad y la textura de la espuma



CAPÍTULO VI: CONCLUSIONES Y RECOMENDACIONES

1. Conclusiones

El algoritmo de segmentación propuesto basado en *marker-controlled watershed* permitió identificar los contornos de las burbujas y obtener sus propiedades geométricas, como el área y el centroide. La combinación de técnicas de preprocesamiento, operaciones morfológicas y segmentación posibilitó la medición del tamaño de las burbujas en cm^2 y $píxeles^2$, así como la construcción de la distribución del tamaño de burbujas (BSD), constituyendo la base para la estimación de la estabilidad y velocidad de las burbujas. Asimismo, la evaluación del BSD, tomando como referencia la segmentación manual de Wright (1999), evidenció que el método de segmentación propuesto logró una representación más precisa de la distribución del tamaño de burbujas que el método de segmentación basado en *fast watershed* desarrollado por Wright al obtener menores errores en todas las métricas evaluadas. Se registraron reducciones de 26.14 % en el MAE, 72.77 % en el MSE, 47.82% en el RMSE y 14.60 % en la distancia de Wasserstein, evidenciando una menor discrepancia respecto al BSD de referencia y una representación más fiel de la distribución del tamaño de burbujas. Asimismo, el MAE resultó ser la métrica más robusta para evaluar el error global del BSD debido a su interpretación directa en número de burbujas y a su menor sensibilidad frente a discrepancias extremas ocasionadas por el incremento de burbujas pequeñas en el primer bin, así como por efectos de sobresegmentación y subsegmentación.

Se implementó un algoritmo basado en el seguimiento de centroides de burbujas estables entre pares consecutivos de frames, permitiendo medir la velocidad de la espuma en cm/s. Los resultados mostraron una velocidad promedio de 22.67 cm/s, demostrando la capacidad del algoritmo para cuantificar la movilidad de la espuma. Asimismo, el análisis de las muestras evaluadas mostró que el 73.33 % de las burbujas permanecieron estables entre frames consecutivos, mientras que los fenómenos de colapso y coalescencia representaron el 13.33 % cada uno.

Adicionalmente, la estabilidad presentó variaciones entre 73 % y 82 % durante el período analizado, según las tendencias registradas en el SCADA.

Como aporte metodológico de la investigación, se clasificaron experimentalmente las distribuciones del tamaño de burbuja (BSD) en cuatro categorías (A, B, C y D), asociadas a diferentes tamaños predominantes de burbuja. Asimismo, se establecieron umbrales experimentales de estabilidad a partir del porcentaje de estabilidad medido, proporcionando criterios preliminares para la interpretación del estado de la espuma.

El análisis del color y la textura de la espuma evidenció que la disminución del tamaño de las burbujas desplaza la distribución del color de la espuma (FCD) hacia intensidades más altas debido a una mayor reflexión de la luz sobre las burbujas pequeñas, mientras que el incremento de su tamaño tiende a aumentar la energía textural obtenida mediante la matriz GLCM. Estos resultados demuestran que las características visuales de la espuma contienen información relevante para la caracterización de su estructura y comportamiento, permitiendo relacionar cambios en el tamaño de las burbujas con variaciones en el color y la textura de la espuma.

La implementación del SCADA en MATLAB App Designer y del Dashboard IoT basado en ThingSpeak™ permitió analizar y monitorear las características de la espuma desde plataformas locales y remotas. La coincidencia de los valores de área de burbuja, velocidad, estabilidad y textura registrados en ambos sistemas confirmó la trazabilidad e integridad de los datos durante las etapas de adquisición, procesamiento, transmisión y visualización, validando el correcto funcionamiento del sistema de monitoreo desarrollado. Las tendencias mostraron valores de estabilidad, velocidad, área y textura entre rangos aproximados de 73 % a 82 %, de 10 a 20 cm/s, de 4 a 7 cm^2 y de 0.13 a 0.16, respectivamente. Asimismo, el análisis de las tendencias de las variables de la espuma evidenció una relación inversa entre el tamaño de burbuja y la velocidad, así como entre el tamaño de burbuja y la estabilidad de la espuma. Del mismo modo, se observó una relación directa entre el tamaño de burbuja y la energía

textural. Estos resultados evidencian que las características físicas, dinámicas y estáticas de la espuma presentan un comportamiento interrelacionado.

Los tiempos computacionales obtenidos demostraron la viabilidad del sistema desarrollado para el procesamiento y monitoreo periódico de las características de la espuma. El cálculo del tamaño de burbuja, la estabilidad y la velocidad requirió aproximadamente 0.41 s por cada par de frames, mientras que el procesamiento de una muestra completa presentó tiempos entre 5.58 y 6.62 s. Asimismo, la actualización de las variables en el SCADA y el Dashboard IoT se realizó en intervalos cercanos a 5 s, registrándose un tiempo adicional de aproximadamente 0.5 s para la transmisión de los datos desde el SCADA hacia la plataforma ThingSpeak™. Estos resultados permitieron la supervisión periódica de las condiciones de la espuma mediante herramientas de monitoreo local y remoto.

Se logró segmentar las imágenes de la espuma y medir sus características físicas, estáticas y dinámicas mediante algoritmos de procesamiento de imágenes, integrando exitosamente la información en un SCADA desarrollado con MATLAB App Designer y en un Dashboard IoT basado en ThingSpeak™ para su análisis y monitoreo. Los resultados obtenidos demostraron la viabilidad del análisis y monitoreo remoto de estas variables desde dispositivos móviles, respaldando la hipótesis planteada y evidenciando la aplicación de tecnologías asociadas a la Industria 4.0. Asimismo, debido a que las investigaciones relacionadas con la visualización de variables de la espuma mediante plataformas IoT son escasas y los datos operacionales mineros suelen ser de acceso restringido, la propuesta desarrollada constituye un aporte para futuras aplicaciones de monitoreo en procesos de flotación.

2. Recomendaciones

Para futuras implementaciones industriales, resulta conveniente incorporar una cámara instalada en línea junto a la celda de flotación para adquirir imágenes de manera continua y automatizar el monitoreo de las características de la espuma.

Asimismo, sería conveniente ampliar la base de imágenes y videos provenientes de diferentes condiciones operacionales con la finalidad de evaluar la robustez de los algoritmos de segmentación en distintos escenarios de flotación. Además, para aplicaciones con mayores tasas de adquisición de imágenes, conviene considerar la optimización del código y la selección adecuada del hardware, incluyendo CPU, GPU, cámara e interfaces de comunicación de alta velocidad como GigE o USB 3.0, junto con un sistema de iluminación apropiado para la región de interés.

La evaluación de la estabilidad de la espuma podría extenderse a períodos de monitoreo más prolongados y a diferentes condiciones operacionales de flotación, con el fin de analizar la influencia de variables como el caudal de aire, la dosificación de reactivos, el pH y la densidad de pulpa sobre los fenómenos de estabilidad, colapso y coalescencia. Asimismo, futuras investigaciones podrían incorporar algoritmos capaces de identificar fenómenos adicionales, tales como el colapso y la coalescencia de burbujas, que actualmente no son considerados en la evaluación de la estabilidad.

Conviene analizar el desempeño del algoritmo de velocidad utilizando un mayor número de videos y condiciones operacionales, a fin de determinar su sensibilidad frente a variaciones en el tamaño de burbuja, el flujo de aire y la estabilidad de la espuma.

Los parámetros derivados de la distribución del color de la espuma (FCD) podrían emplearse como variables de entrada para sistemas de monitoreo inteligente o modelos predictivos orientados a la estimación indirecta de cambios en el tamaño de burbuja durante el proceso de flotación. Del mismo modo, futuros estudios podrían evaluar otras propiedades de la matriz GLCM, tales como el contraste, la correlación y la homogeneidad, para determinar cuál presenta una mayor sensibilidad en la caracterización de cambios estructurales de la espuma bajo diferentes condiciones de operación.

Se recomienda evaluar el sistema de monitoreo en un entorno industrial con adquisición continua de imágenes y un mayor número de muestras, a fin de validar su

desempeño bajo condiciones reales de operación y comunicación. Asimismo, conviene ampliar el análisis estadístico de las relaciones entre las variables de la espuma mediante un mayor número de muestras, con el objetivo de cuantificar dichas correlaciones y evaluar su aplicabilidad en estrategias de supervisión y control del proceso de flotación. Adicionalmente, la visualización remota del BSD en el Dashboard IoT podría complementarse mediante el envío de imágenes segmentadas a la plataforma ThingSpeak™, aprovechando sus herramientas de análisis para la generación y visualización automática de histogramas de la distribución del tamaño de burbujas (BSD).

Con el objetivo de incrementar la frecuencia de actualización de las variables monitoreadas, resulta conveniente optimizar las etapas de segmentación y visualización de imágenes para reducir los tiempos computacionales y evaluar la implementación del sistema con adquisición continua de imágenes. Asimismo, futuros trabajos podrían investigar estrategias para reducir los efectos de sobresegmentación y subsegmentación mediante el ajuste adaptativo del parámetro H de la transformada de máximos extendidos o mediante técnicas avanzadas de segmentación.

Las variables obtenidas mediante el sistema desarrollado constituyen una base potencial para el desarrollo de modelos de aprendizaje automático o redes neuronales orientados a la predicción de indicadores metalúrgicos. Asimismo, la información generada podría emplearse en el diseño de estrategias de control experto o control avanzado que permitan optimizar las condiciones de operación de las celdas de flotación. Finalmente, futuros trabajos podrían evaluar algoritmos de segmentación más avanzados para mejorar la estimación del BSD, tales como Semantic Segmentation + Marker-Based Watershed, YOLO-Seg, MF-UNet++ + Watershed, Mask R-CNN, Nested U-Net y U-Net++, así como analizar su compromiso entre exactitud y tiempo computacional.

REFERENCIAS BIBLIOGRÁFICAS

Wright, B. A. (1999). *The Development of a vision-based flotation froth analysis system* [Master Thesis, University of Cape Town]. OpenUCT Institutional Repository. http://hdl.handle.net/11427/9923

Bravo Gálvez, A. C. (2004). *Manual de flotación de minerales.* [Archivo PDF]. Academia.edu. https://www.academia.edu/11361321/Man_de_Flotación_M_2004

Chai, T. y Draxler, R. R. (2014, 30 de junio). Root mean square error (RMSE) or mean absolute error (MAE)? Arguments against avoiding RMSE in the literature. *Geoscientific Model Development*. *7*(3), 1247-1250. https://doi.org/10.5194/gmd-7-1247-2014

De Nardis, L., Mohammadpour, A., Caso, G., Usman, A. y Di Benedetto, M-G. (2022, 19 de febrero). Internet of things platforms for academic research and development: A critical review. *Applied Sciences, 12*(4), 2172, 1-19. https://doi.org/10.3390/app12042172

Farrokhpay, S. (2011, 16 de marzo). The significance of froth stability in mineral flotation: A review. *Advances in Colloid and Interface Science. 166*(1-2), 1-7. https://doi.org/10.1016/j.cis.2011.03.001

Forbes, G. (2007). *Texture and bubble size measurements for modelling concentrate grade in flotation froth system*. [doctoral thesis, University of Cape Town]. OpenUCT Institutional Repository. http://hdl.handle.net/11427/9247

Francis, J. J. (2001). *Machine vision for froth flotation* [unpublished doctoral thesis]. Department of Electrical Engineering, University of Cape Town.

Gonzales, R. C., Woods, R. E. y Eddins, S. L. (2004). *Digital image processing using MATLAB*. Pearson.

Hadler, K., Smith, C. D. y Cilliers, J. J. (2010, octubre). Recovery vs. mass pull: The link to air recovery. *Minerals Engineering, 23(*11-13*)*, 994-1002. https://doi.org/10.1016/j.mineng.2010.04.007

Hodson, T. O. (2022, 19 de julio). Root-mean-square error (RMSE) or mean absolute error (MAE): when to use them or not. *Geoscientific Model Development, 15*(14), 5481-5487. https://doi.org/10.5194/gmd-15-5481-2022

Hosseini, M. R., Haji Amin Shirazi, H., Massinaei, M. y Mehrshad, N. (2015, 26 de febrero). Modeling the relationship between froth bubble size and flotation performance using image analysis and neural networks. *Chemical Engineering Communications, 202*(7), 911-919. https://doi.org/10.1080/00986445.2014.886201

Kuňa, V., Balberčák, J., Boháček, Š. y Ihnát, V. (2021, abril). Elimination of adhesive impurities of the recovered paper in flotation process. Wood Research. 66(2), 221-230. https://doi.org/10.37763/wr.1336-4561/66.2.221230

Lin, B., Recke, B., Knudsen, J. K.H. y Jørgensen, S. B. (2008, junio). Bubble size estimation for flotation processes. *Minerals Engineering, 21(*7*)*, 539-548. https://doi.org/10.1016/j.mineng.2007.11.004

Massinaei, M., Jahedsaravani, A., Taheri, E. y Khalilpour, J. (2019, 1 de febrero). Machine vision based monitoring and analysis of a coal column flotation circuit. *Powder Technology, 343*, 330-341. https://doi.org/10.1016/j.powtec.2018.11.056

Mesa, D., Kracht, W. y Díaz, G. (2016, noviembre). Textural image classification of foams based on variographic analysis. *Minerals Engineering. 98*, 52-59. https://doi.org/10.1016/j.mineng.2016.07.012

Nad, A., Jooshaki, M., Tuominen, E., Michaux, S., Kirpala, A. y Newcomb, J. (2022, 7 de febrero). Internet of things platforms for academic research and development: A critical review. *Minerals, 12*(2), 210, 1-21. https://doi.org/10.3390/min12020210

Ni, K., Bresson, X., Chan, T. y Esedoglu, S. (2009, 4 de abril). Local histogram based segmentation using the wasserstein distance. *International Journal of Computer Vision. 84*(1), 97-111. https://doi.org/10.1007/s11263-009-0234-0

Ostadrahimia, M., Farrokhpayb, S., Gharibia, K. y Dehghania, A. (2020, 5 de junio). A new empirical model to calculate bubble size in froth flotation columns. *Colloids and Surfaces A. 594*, 124672, 1-5. https://doi.org/10.1016/j.colsurfa.2020.124672

Rendon Montoya, A., Dávila Peralta, I. D. y Flores Guzman,. N. (2016). *Sistema analizador de burbujas en celdas de flotación basado en visión artificial* (Patente internacional n.º WO 2016/099243 A1). Organización Mundial de la Propiedad Intelectual. https://patents.google.com/patent/WO2016099243A1

Sadr-Kazemi, N. y Cilliers, J.J. (1998, 15 de junio). An image processing algorithm for measurement of flotation froth bubble size and shape distributions. *Minerals Engineering, 10(10)*, 1075-1083. https://doi.org/10.1016/S0892-6875(97)00094-0

Jämsä-Jounela, S.-L. (2019). Future automation systems in context of process systems and minerals engineering. *IFAC-PapersOnLine, 52(*25*)*, 403-408. https://doi.org/10.1016/j.ifacol.2019.12.570

Supomo, A., Yap, E., Zheng, X., Banini, G., Mosher, J. y Partanen, A. (2008, noviembre). PT Freeport Indonesia's mass-pull control strategy for rougher flotation. *Minerals Engineering*, *21*(12-14), 808-816. https://doi.org/10.1016/j.mineng.2008.07.002

The MathWorks, Inc. (2025). *Propiedades de la matriz de coocurrencia de nivel de gris (GLCM).* MATLAB Help Center. https://la.mathworks.com/help/images/ref/graycoprops.html

The MathWorks, Inc. (2025a). *Marker-controlled watershed segmentation.* MATLAB Help Center. https://www.mathworks.com/help/images/marker-controlled-watershed-segmentation.html

The MathWorks, Inc. (2025b). *Overview.* ThingSpeak. https://la.mathworks.com/products/thingspeak.html

The MathWorks, Inc. (2025c). *Student License.* ThingSpeak. https://thingspeak.mathworks.com/prices/thingspeak_student

The MathWorks, Inc. (2026). Sign In - ThingSpeak IoT. https://thingspeak.mathworks.com/login

Unzueta Arce, R. (2016). *Sistema de medición basado en visión artificial para circuito rougher de laboratorio.* [Tesis de pregrado, Universidad Técnica Federico Santa María]. Repositorio USM. https://doi.org/10.71700/dspace-memorias/1972

Sbárbaro, D. y Del Villar, R. (2010). *Advanced control and supervision of mineral processing plants.* Springer.

Zezulka, F., Marcon, P., Bradac, Z., Arm, J., Benesl, T. y Vesely, I. (2018). Communication systems for Industry 4.0 and the IIoT. *IFAC-PapersOnLine, 51(6)*, 2, 150-155. https://doi.org/10.1016/j.ifacol.2018.07.145

Zhang, J., Tang, Z., Ai, M. y Gui, W. (2018, 13 de junio). Fuzzy association rule based froth surface behavior control in zinc froth flotation. *Symmetry, 10(6)*, 216, 1-17. https://doi.org/10.3390/sym10060216

ANEXOS

1. Los códigos y la base de datos de las imágenes y videos

Los códigos y la base de datos tales como imágenes videos etc., se encuentran en un repositorio de GitHub:

https://github.com/AlexTapiaM/IoT-Dashboard-for-Froth-Characteristics.git

2. Contactos para solicitar la base de datos

Para revisiones pueden solicitarlo a los siguientes correos:

gerardo.tapia@ucsm.edu.pe

gerardo.alexander.tapia@gmail.com