



Desarrollo de un procedimiento de calibración de sonómetros y calibradores acústicos de conformidad con las normas IEC 61672-3 e IEC 60942

Juan Felipe Maldonado Pedraza

Universidad Nacional de Colombia
Facultad de Ingeniería, Departamento de Ingeniería Eléctrica y Electrónica
Bogotá, Colombia
2021

Desarrollo de un procedimiento de calibración de sonómetros y calibradores acústicos de conformidad con las normas IEC 61672-3 e IEC 60942

Juan Felipe Maldonado Pedraza

Trabajo de grado presentado como requisito parcial para optar al título de:
Magister en Automatización Industrial

Director:
Ph.D. Leonardo Enrique Bermeo Clavijo

Línea de investigación:
Automatización en procesos de metrología
Universidad Nacional de Colombia
Facultad de Ingeniería
Bogotá, Colombia
2022

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Lorem ipsum dolor sit amet, consectetur adipiscing elit.

Agradecimientos

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Resumen

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Palabras clave: Calibración, metrología, sonómetros, calibradores acústicos, automatización, visión de máquina, cadena de Markov

Abstract

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Keywords: Calibration, metrology, sound level meters, acoustic calibrators, automation, computer vision, Markov chain

Índice general

Agradecimientos	VII
Resumen	IX
Lista de figuras	XIII
Lista de tablas	1
1. Introducción	2
1.1. Planteamiento del problema	2
1.2. Objetivos	3
1.2.1. General	3
1.2.2. Específicos	3
1.2.3. Alcance de los objetivos	3
1.2.4. Antecedentes	4
2. Metodología e instrumentación	5
2.1. Instrumentos bajo calibración	5
2.1.1. Calibradores acústicos	5
2.1.2. Sonómetros integradores	6
2.2. Métodos normalizados	9
2.2.1. Descripción general de la calibración periódica de calibradores acústicos de acuerdo con la IEC 60942:2017	9
2.2.2. Descripción general de las pruebas periódicas seleccionadas de acuerdo con la IEC 61672-3:2013	9
2.3. Instrumentación	9
2.3.1. Patrones e instrumentos para la calibración periódica de calibradores acústicos	9
2.3.2. Patrones e instrumentos para la calibración periódica de sonómetros	9
2.3.3. Comandos SCPI	9
3. Sistema de reconocimiento de imágenes para el valor de medición	10
3.1. Algoritmo de reconocimiento de caracteres	10

3.2. Filtro gaussiano y <i>padding</i>	10
3.3. Segmentación	13
3.4. Descriptor local SIFT	15
3.5. Reducción de dimensionalidad por análisis de componentes principales (KPCA)	17
3.6. Clasificador bayesiano normal	19
3.7. Resultados	22
3.8. Discusión	27
4. Implementación de los procedimientos de calibración	28
4.1. Automatización de la calibración periódica de calibradores acústicos	28
4.1.1. GRAFCET descriptivo del proceso	28
4.1.2. Implementación en Python	28
4.2. Automatización de la calibración periódica de sonómetros	28
4.2.1. Implementación en Python	28
5. Modelamiento de incertidumbre de medición en la calibración periódica de sonómetros	29
5.1. Cadenas de Markov de tiempo discreto	30
5.1.1. Matriz de probabilidades de transición	30
5.1.2. Valor esperado	30
5.1.3. Probabilidad estacionaria	30
5.2. Incertidumbre de medición	30
5.2.1. Incertidumbre tipo A	30
5.2.2. Incertidumbre tipo B	30
5.2.3. Incertidumbre típica combinada	30
5.2.4. Grados efectivos de libertad, factor de cobertura e incertidumbre expandida	30
5.3. Algoritmo para creación de estados y tasas de transición de la cadena de Markov	30
5.3.1. Implementación de la cadena de Markov en Python	30
6. Conclusiones y recomendaciones	31
6.1. Conclusiones	31
6.2. Recomendaciones	31
A. Anexo: Certificado de calibración de calibrador acústico	32
B. Anexo: Certificado de calibración de sonómetro	33
Bibliografía	34

Índice de figuras

2-1. Calibrador acústico multifunción Brüel & Kjær 4226.	5
2-2. Calibrador acústico Brüel & Kjær 4231 acoplado al micrófono de un sonómetro Brüel & Kjær 2250.	6
2-3. Configuraciones de <i>hardware</i> del sonómetro Brüel & Kjær 2250.	7
3-1. Diagrama de flujo del algoritmo de reconocimiento de caracteres	11
3-2. Ejemplo de una fotografía de la pantalla de un sonómetro 01dB CUBE.	22
3-3. Puntos clave para el cálculo del descriptor SIFT de una imagen de un dígito.	22
3-4. Resultados de procesamiento para tres muestras del conjunto de entrenamiento.	24
3-5. Vectores de características de las tres muestras de entrenamiento procesadas.	25
3-6. Matriz de confusión del clasificador bayesiano propuesto.	26

Índice de cuadros

1. Introducción

1.1. Planteamiento del problema

En la actualidad, la creciente contaminación acústica amerita la implementación de redes de monitoreo continuo de ruido o mediciones puntuales empleando instrumentos adecuados como son los sonómetros con el propósito de cuantizar los niveles de ruido ambiental, de emisión de ruido de fuentes sonoras específicas y de exposición sonora, para luego comparar con los niveles máximos permitidos por la normativa relacionada nacional e internacional y tomar decisiones al respecto. Para garantizar la confiabilidad de tales mediciones o asegurar la validez de sus resultados, en Colombia, entidades como el Instituto de Hidrología, Meteorología y Estudios Ambientales (IDEAM) exigen que las organizaciones que prestan estos servicios cuenten con sonómetros calibrados periódicamente bajo el estándar internacional IEC 61672-3:2013. por parte de un organismo de evaluación de la conformidad (OEC), en este caso un laboratorio de calibración acreditado por el Organismo Nacional de Acreditación de Colombia (ONAC) bajo el estándar ISO 17025, con el fin de verificar que estos instrumentos continúan cumpliendo las especificaciones normalizadas según su clase. La norma IEC 61672-3 describe una serie de pruebas acústicas y eléctricas que se realizan a sonómetros integradores clase 1 y 2, cuyo propósito es comprobar el funcionamiento del sonómetro en: 1) La sensibilidad de su micrófono (para lo cual se usa un calibrador acústico calibrado previamente y que esté en conformidad con las especificaciones de la IEC 60942). 2) Las redes de ponderación frecuencial A, C y Z. 3) En las ponderaciones temporales F (*fast*) y S (*slow*). 4) En el rango lineal. 5) En la medición de niveles promediados en el tiempo, niveles de exposición sonora y niveles pico. 6) En la indicación de sobrecarga. 7) En la exposición a largos periodos de medición y a niveles de sonido elevados. Dicha comprobación se hace comparando con las especificaciones definidas en la norma IEC 61672-1:2013.

1.2. Objetivos

1.2.1. General

Desarrollar un sistema de calibración periódica de sonómetros y calibradores acústicos de conformidad con las normas IEC 61672-3:2013 e IEC 60942:2017.

1.2.2. Específicos

1. Formular un modelo en GRAFCET como base para el desarrollo de un sistema de calibración periódica de calibradores acústicos.
2. Implementar las secuencias de comando (a través de bus GPIB) para configurar parámetros de señal y, a su vez, recibir resultados de los instrumentos de medición.
3. Desarrollar un método de reconocimiento de imágenes para detectar los niveles instantáneos ponderados en tiempo y en frecuencia desde la pantalla del sonómetro.
4. Desarrollar un método que permita tener en cuenta la variabilidad de los niveles en pantalla instantáneos ponderados en tiempo y en frecuencia del objetivo 3, (mediante mediciones de larga duración), para la estimación del mesurando y de la incertidumbre de medición.

1.2.3. Alcance de los objetivos

El sistema de calibración se implementará para ejecutar las pruebas de calibración de los numerales 9.3 (apoyado en la IEC 60942), 13, 14 y 16 de la IEC 61672-3. Los indicadores de interés serán los niveles instantáneos con ponderación temporal (*slow* o *fast*) y ponderación frecuencial (*A*, *C*, o *Z*), i.e. $L_{AF}, L_{AS}, L_{CF}, L_{CS}, L_{ZF}$ o L_{ZS} , dependiendo de la prueba y según estén disponibles en el sonómetro sujetos al periodo de actualización de la pantalla del sonómetro. El sistema tendrá en cuenta el modelo del proceso estocástico para la estimación de incertidumbre expandida de medición.

1.2.4. Antecedentes

Sistemas de calibración comerciales desarrollados por fabricantes

Sistemas de calibración desarrollados por otras organizaciones

2. Metodología e instrumentación

2.1. Instrumentos bajo calibración

2.1.1. Calibradores acústicos

De acuerdo con la normativa internacional, un calibrador acústico es un dispositivo diseñado para producir uno o más niveles de presión sonora conocidos (en dB referenciados a $20 \mu\text{Pa}$) a una o más frecuencias especificadas (en Hz) cuando se acopla a modelos específicos de micrófono en configuraciones específicas (IEC-TC29, 2017). Normalmente, la señal senoidal generada por algún transductor, como un altavoz o, en el caso de los pistófonos, un pistón mecánico cuyo movimiento genera en la cavidad una velocidad de volumen conocida. Como ejemplo, en la figura 2-1 se muestra un calibrador acústico multifunción usado como referencia en muchos laboratorios: el Brüel & Kjær 4226, que es capaz de generar 94 dB, 104 dB y 114 dB en las frecuencias de octava desde 31,5 Hz hasta 16 kHz, más la frecuencia de 12,5 Hz.

Figura 2-1.: Calibrador acústico multifunción Brüel & Kjær 4226.



Tomado de <https://www.transcat.com/bruel-kjaer-4226-acoustic-calibrator-94-104-and-114db-used>

Generalmente los calibradores acústicos son empleados para determinar la sensibilidad en campo de presión (típicamente en mV/Pa o en dB referenciados a 1 V) de modelos especificados de micrófonos en configuraciones dadas, pero también es utilizado para verificar o ajustar la sensi-

bilidad de algún dispositivo o sistema de medición acústica. Un ejemplo de calibrador acoplado para comprobar la indicación de un sonómetro se muestra en la figura 2-2.

Figura 2-2.: Calibrador acústico Brüel & Kjær 4231 acoplado al micrófono de un sonómetro Brüel & Kjær 2250.



Tomado de <https://www.bksv.com/en/knowledge/blog/sound/getting-started-sound-level-meter>

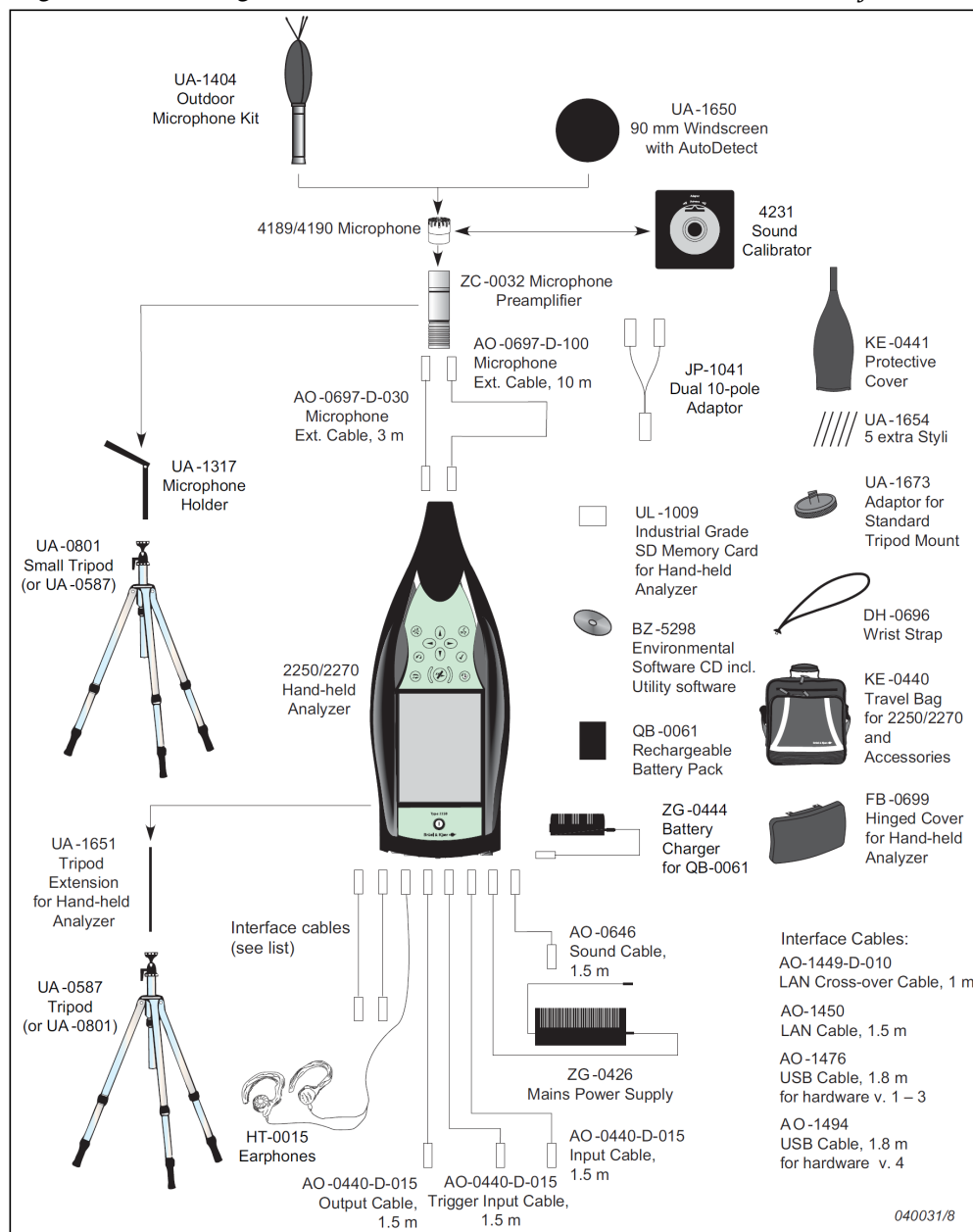
La norma IEC 60942:2017 establece una clasificación de los calibradores según sus especificaciones (límites de aceptación), de la más a la menos restrictiva: Clase LS (*laboratory standard*), clase 1 o clase 2. La comprobación de que cierto modelo de calibrador cumple con todas las especificaciones normalizadas según su clase la realiza una organización independiente acreditada para hacer pruebas de aprobación de modelo de acuerdo con los lineamientos del Anexo A de la IEC 60942:2017. Pero además, un usuario de un calibrador acústico debería calibrar periódicamente su instrumento para garantizar la trazabilidad a los estándares nacionales y la confiabilidad de sus resultados. Esta calibración periódica es llevada a cabo por organismos evaluadores de la conformidad acreditados en ISO 17025 para realizar los ensayos periódicos de acuerdo con el Anexo B de la IEC 60942:2017. Los ensayos periódicos son el objeto de automatización en este proyecto y se describen en la sección 2.2.1.

2.1.2. Sonómetros integradores

Brüel & Kjær, uno de los fabricantes más prominentes de sonómetros define consistentemente los conceptos básicos sobre dichos instrumentos en uno de sus artículos (B&K, 2021). Básicamente, un sonómetro es un instrumento diseñado para medir niveles de sonido de una forma estandarizada; su respuesta al sonido se asemeja a la del oído humano y proporciona medidas de niveles de presión sonora objetivas y reproducibles. Generalmente, los sonómetros son empleados en el monitoreo de ruido proveniente de diversas fuentes sonoras, como plantas industriales, tráfico rodado, aeronáutico o ferroviario, conciertos, etc. Como se puede ver en la figura 2-3, un sonómetro típico consta de un micrófono, un preamplificador, una unidad de procesamiento de señal (interna) y una pantalla. Regularmente el preamplificador hace parte del cuerpo del sonómetro, pero no siempre es el caso; un sonómetro podría estar provisto de

cables de extensión que separen el preamplificador de la unidad de procesamiento.

Figura 2-3.: Configuraciones de *hardware* del sonómetro Brüel & Kjær 2250.



Tomado de del Manual de Instrucciones (B&K, 2016)

En cuanto al flujo de señal, el micrófono es un transductor electroacústico que transforma la señal acústica en una señal eléctrica. La mayoría de los micrófonos empleados en mediciones acústicas son de condensador, y gracias a su construcción es el mejor tipo para garantizar precisión, estabilidad y confiabilidad en los resultados. No obstante, la señal eléctrica proporcionada por un micrófono es de baja amplitud (aún con micrófonos de alta gama cuya sensibilidad se

encuentra típicamente en el orden de los 50 mV/Pa , por lo que se requiere una amplificación para que la unidad de procesamiento manipule la señal en un nivel adecuado, este es el objetivo del preamplificador. Luego, en la unidad de procesamiento se ejecutan diferentes cálculos a partir de la señal, los mínimos requeridos por la norma internacional IEC 61672-1:2013 y utilizados en este proyecto son:

- **Ponderación temporal:** Modifica la respuesta temporal del sonómetro frente a las variaciones en el nivel de presión sonora.

2.2. Métodos normalizados

2.2.1. Descripción general de la calibración periódica de calibradores acústicos de acuerdo con la IEC 60942:2017

2.2.2. Descripción general de las pruebas periódicas seleccionadas de acuerdo con la IEC 61672-3:2013

Indicación a la frecuencia de comprobación de la calibración

Ponderaciones frecuenciales y temporales a 1 kHz

Ponderaciones frecuenciales con señales eléctricas

Linealidad de nivel en el rango de niveles de referencia

2.3. Instrumentación

2.3.1. Patrones e instrumentos para la calibración periódica de calibradores acústicos

2.3.2. Patrones e instrumentos para la calibración periódica de sonómetros

2.3.3. Comandos SCPI

3. Sistema de reconocimiento de imágenes para el valor de medición

3.1. Algoritmo de reconocimiento de caracteres

De manera general, la solución propuesta, para el reconocimiento del valor de medición indicado en la pantalla del sonómetro, consta de varios pasos que se presentan en el diagrama de la figura 3-1.

Concerniente al procesamiento y reconocimiento de imágenes, a continuación se presenta un breve marco de referencia teórico de los principales bloques de proceso del diagrama de flujo de la 3-1, tomando como guía el libro de Robert Szeliski Richard (2011).

3.2. Filtro gaussiano y padding

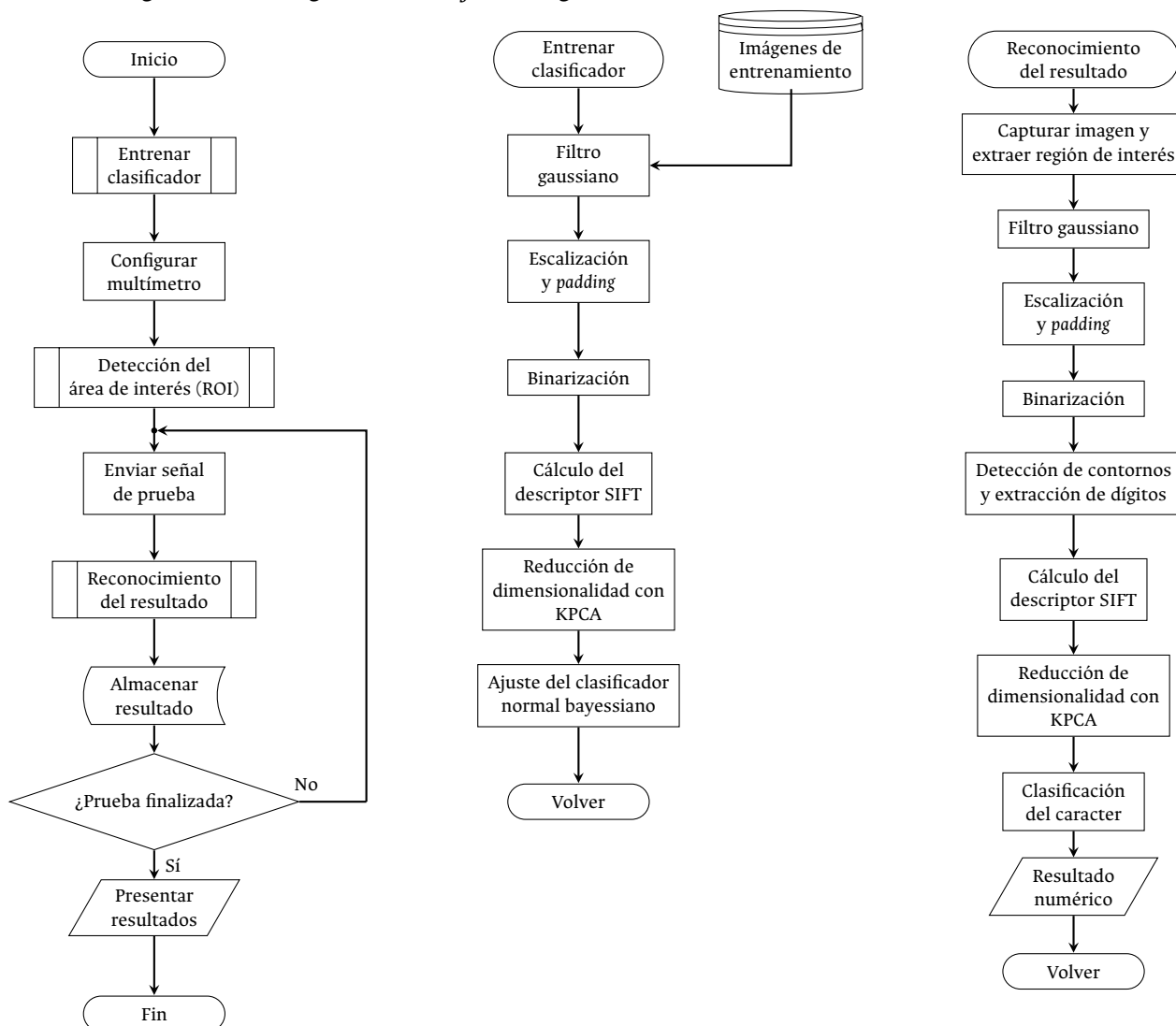
También llamado filtro de suavizado o de desenfoque, es una operación local en imágenes bi-dimensionales, pues se efectúa en vecindarios con un tamaño determinado en píxeles; el valor final de un píxel depende de los valores de los píxeles que pertenecen a su correspondiente vecindario y, como en todos los casos de filtros lineales, de una función de ponderación. Esta operación local viene a ser la de correlación: la suma ponderada de los píxeles de entrada, como se presenta en las ecuaciones (3-1a) y (3-1b).

$$g(i, j) = \sum_{k, l} f(i + k, j + l) h(k, l) \quad (3-1a)$$

$$g = f \otimes h \quad (3-1b)$$

Donde f es la imagen de entrada, g la imagen de salida y h la máscara o *kernel*, que contiene los coeficientes del filtro. La operación contraparte sería la convolución, en la que se usa el kernel invertido, como se expresa en las ecuaciones (3-2a) y (3-2b).

Figura 3-1.: Diagrama de flujo del algoritmo de reconocimiento de caracteres



$$g(i, j) = \sum_{k, l} f(k, l) h(i - k, j - l) \quad (3-2a)$$

$$g = f * h \quad (3-2b)$$

En la convolución, h es la respuesta al impulso, ya que si se convoluciona la máscara h con una señal impulsiva $\delta(i, j)$, se obtiene la misma máscara ($h * \delta = h$).

Para el caso del filtro gaussiano, el *kernel* se obtiene a partir de la típica función exponencial de Gauss. Particularmente, considerando la implementación de `OpenCV` que se ejecuta con la instrucción `GaussianBlur()`, la función sería la bivariada no correlacionada que se presenta en la ecuación 3-3.

$$h(x, y) = \frac{1}{2\pi\sigma_x\sigma_y} \exp\left(-\frac{(x - \mu_x)^2}{2\sigma_x^2} - \frac{(y - \mu_y)^2}{2\sigma_y^2}\right) \quad (3-3)$$

Además, la función de `OpenCV` permite especificar un tamaño definido de ventana y esta calculará automáticamente las varianzas como se muestra en las ecuaciones (3-4a) y (3-4b).

$$\sigma_x = \left(\frac{n_x - 1}{2}\right) 0.3 + 0.8, \quad n_x = \text{ancho} - 1 \quad (3-4a)$$

$$\sigma_y = \left(\frac{n_y - 1}{2}\right) 0.3 + 0.8, \quad n_y = \text{alto} - 1 \quad (3-4b)$$

Por supuesto, en la práctica, esta operación requiere que se agreguen píxeles en los bordes de la imagen original, según el tamaño determinado de la ventana; a esto se le conoce como *padding* y es una operación que realiza por defecto `OpenCV`, con la que es posible elegir el modo del *padding*. Existen diferentes modos de *padding*, de los cuales se describen los siguientes:

- *zero*: Todos los píxeles añadidos se establecen en 0.
- *constant*: Todos los píxeles añadidos se establecen en un valor determinado, lógicamente, según el tipo de dato o profundidad de bits de la imagen digitalizada.
- *clamp*: Repite los píxeles de los bordes indefinidamente.
- *cyclic*: Repite la imagen como si fuese un mosaico.
- *mirror*: Refleja los últimos píxeles del borde.
- *extend*: Los píxeles añadidos corresponden a la señal reflejada substraída de los bordes.

3.3. Segmentación

Existen diferentes técnicas de segmentación: basada en umbrales, en bordes, en regiones, por agrupación o por *matching*. Por los propósitos del algoritmo, es suficiente con una umbralización sencilla para discriminar los segmentos de la imagen, que corresponderían a los dígitos indicados en la pantalla del sonómetro. Sin embargo, el umbral no puede ser el mismo para todas las imágenes, pues cada una puede estar influenciada por los efectos de iluminación, áreas o colores de los segmentos. Así que una forma sistemática de determinar el umbral es empleando el algoritmo de Otsu (1979), que básicamente busca maximizar la separación entre las clases de niveles de grises del histograma de la imagen usando los momentos de cero y primer orden.

La formulación de Otsu es un método no supervisado basado en el análisis discriminante para evaluar la bondad del umbral y seleccionar automáticamente un límite óptimo. En primer lugar, Otsu normaliza el histograma de niveles de grises y lo considera como una distribución de probabilidad. Si hay L niveles de grises, entonces el número de píxeles en la imagen es $N = n_1 + n_2 + \dots + n_L$; n_i es el número de píxeles que tienen un nivel i . Luego, la distribución de probabilidad queda expresada como:

$$p_i = \frac{n_i}{N}, \quad p_i \geq 0, \quad \sum_{i=1}^L p_i = 1 \quad (3-5)$$

Ahora, se buscan dos clases C_0 y C_1 , que corresponden a los píxeles que pertenecen al fondo y a los que pertenecen a los objetos, separados por el nivel k . Las probabilidades de cada clase se definen intuitivamente en las ecuaciones (3-6a) y (3-6b).

$$\omega_0 = \Pr(C_0) = \sum_{i=1}^k p_i = \omega(k) \quad (3-6a)$$

$$\omega_1 = \Pr(C_1) = 1 - \omega(k) \quad (3-6b)$$

Luego, los valores esperados condicionales de cada clase son:

$$\mu_0 = \sum_{i=1}^k i \Pr(i|C_0) = \sum_{i=1}^k i \frac{p_i}{\omega_0} = \frac{\mu(k)}{\omega(k)} \quad (3-7a)$$

$$\mu_1 = \sum_{i=k+1}^L i \Pr(i|C_1) = \sum_{i=k+1}^L i \frac{p_i}{\omega_1} = \frac{\mu_T - \mu(k)}{1 - \omega(k)} \quad (3-7b)$$

Donde $\omega(k)$ y $\mu(k)$ son los momentos acumulados de cero y primer orden correspondientemente hasta el k -ésimo nivel, y están definidos como en las ecuaciones (3-8).

$$\omega(k) = \sum_{i=1}^k p_i \quad (3-8a)$$

$$\mu(k) = \sum_{i=1}^k i p_i \quad (3-8b)$$

Similarmente, μ_T es la media total de los niveles de grises de la imagen original y σ_T^2 la varianza total, i.e. los momentos de primer y segundo orden de la imagen, definidos como:

$$\mu_T = \sum_{i=1}^L i p_i \quad (3-9a)$$

$$\sigma_T^2 = \sum_{i=1}^L (i - \mu_T)^2 p_i \quad (3-9b)$$

En principio, si las clases están separadas en sus niveles de grises entonces hay una umbralización adecuada; consecuentemente, un umbral que resulte en la mejor separación de clases según sus niveles de grises será un umbral óptimo. Hay por lo menos tres medidas de separación entre clases que se pueden maximizar, pero, por simplicidad (dado que depende de los momentos de orden cero y uno), conviene usar la medida definida en (3-10) como criterio para el análisis discriminante.

$$\eta = \frac{\sigma_B^2}{\sigma_T^2} \quad (3-10)$$

Donde σ_B^2 puede ser denominado como varianza entre clases y se define como:

$$\sigma_B^2 = \omega_0 \omega_1 (\mu_1 - \mu_0)^2 \quad (3-11)$$

Finalmente, condensando las ecuaciones (3-7) a (3-11), el umbral óptimo k^* que maximiza η y, proporcionalmente, σ_B^2 se encuentra con:

$$\sigma_B^2(k^*) = \max_{1 \leq k \leq L} \sigma_B^2(k) \quad (3-12)$$

Donde

$$\sigma_B^2(k) = \frac{[\mu_T \omega(k) - \mu(k)]^2}{\omega(k) [1 - \omega(k)]} \quad (3-13)$$

Esta optimización se puede realizar de forma iterativa con unos valores iniciales de $\omega(0)$ y $\mu(0)$, luego iterando con todos los posibles valores de $k = 0, 1, \dots, L$, y calculando $\sigma_B^2(k)$. El umbral

óptimo k^* será el máximo valor obtenido de $\sigma_B^2(k)$.

3.4. Descriptor local SIFT

Como en esta aplicación particular, cuyo funcionamiento se pretende en tiempo real, el objeto de reconocimiento es simple, no hace falta el sofisticado algoritmo de detección, localización y orientación de puntos característicos de la presentación original del descriptor SIFT que hizo Lowe Lowe (2004). No obstante, la implementación del algoritmo aquí presentada sí tiene en cuenta su propuesta de descriptor local basada en la magnitud y dirección de los gradientes de cada píxel perteneciente a la región alrededor de cada punto característico.

En primer lugar se calcula la magnitud y dirección del gradiente de cada píxel, lo cual finalmente es una valoración del cambio direccional en la intensidad de la imagen. La dirección final del gradiente en un píxel es aquella en la que ocurre el máximo cambio de intensidad, y la magnitud sería el máximo cambio de intensidad. Las diferencias de intensidad en x y en y se determinan con las ecuaciones (3-14a) y (3-14b), luego, la dirección con (3-14c) y la magnitud con (3-14d).

$$\delta_x = I(x + 1, y) - I(x - 1, y) \quad (3-14a)$$

$$\delta_y = I(x, y + 1) - I(x, y - 1) \quad (3-14b)$$

$$\theta(x, y) = \tan^{-1} \left(\frac{\delta_y}{\delta_x} \right) \quad (3-14c)$$

$$|\nabla f(x, y)| = \sqrt{\delta_x^2 + \delta_y^2} \quad (3-14d)$$

Luego de obtener todas las magnitudes y direcciones de los gradientes, se conforma un histograma de orientaciones en 8 intervalos por cada ventana de $n \times n = 4 \times 4$ píxeles; es decir, según la dirección de cada vector, su magnitud se suma en el respectivo intervalo del histograma de la ventana al que pertenece. Finalmente queda un vector de características de 128 valores que es la concatenación de todos los histogramas, cada uno de 8 valores. Pero la contribución de esta magnitud a su intervalo de orientación correspondiente es ponderada por la función gaussiana de la ecuación (3-15) con σ igual a la mitad del ancho de la ventana del descriptor, con el propósito de dar menor peso a los gradientes que están más lejos del centro del descriptor.

$$G(x, y) = \frac{1}{2\pi\sigma^2} \exp \left(-\frac{x^2 + y^2}{2\sigma^2} \right) \quad (3-15)$$

El objetivo del histograma es permitir que hayan cambios locales más grandes en las direcciones de los gradientes pero que contribuyan al mismo intervalo en el histograma. Ahora bien, pue-

den ocurrir cambios abruptos en el histograma cuando en realidad hay cambios suaves en las direcciones de las muestras de gradientes, esto debido a los efectos de los límites en los intervalos. Para mitigar este efecto se aplica una interpolación trilineal para distribuir la magnitud de cada muestra de gradiente en intervalos de histograma adyacentes, en función de la "distancia" de la dirección de la muestra desde el valor central del intervalo; esto queda reflejado en las funciones de ponderación de las ecuaciones (3-16a) a (3-16c).

$$\delta\theta = 360/n$$

$$i\theta(x, y) = \left\lceil \frac{\theta(x, y)}{\delta\theta} \right\rceil$$

$$ix(x, y) = \left\lceil \frac{x}{\delta x} \right\rceil$$

$$iy(x, y) = \left\lceil \frac{y}{\delta y} \right\rceil$$

$$\nabla\theta(x, y) = i\theta(x, y) - \frac{\theta(x, y)}{\delta\theta}$$

$$\nabla x(x, y) = ix(x, y) - \frac{x}{\delta x}$$

$$\nabla y(x, y) = iy(x, y) - \frac{y}{\delta y}$$

$$w_k(x, y) = \begin{cases} \nabla\theta(x, y), & k = i\theta(x, y) \\ 1 - \nabla\theta(x, y), & k = i\theta \text{ mód } 8 + 1 \\ 0, & \text{en caso contrario} \end{cases} \quad (3-16a)$$

$$w_i(x, y) = \begin{cases} \nabla x(x, y), & i = ix(x, y) \\ 1 - \nabla x(x, y), & i = ix(x, y) + 1 \\ 0, & \text{en caso contrario} \end{cases} \quad (3-16b)$$

$$w_j(x, y) = \begin{cases} \nabla y(x, y), & j = iy(x, y) \\ 1 - \nabla y(x, y), & j = iy(x, y) + 1 \\ 0, & \text{en caso contrario} \end{cases} \quad (3-16c)$$

En seguida, se conforma el histograma como se presenta a continuación:

$$H = (H_{11}, H_{12}, \dots, H_{nn})$$

$$H_j = (h_1, h_2, \dots, h_n)$$

$$h_k(x, y) = \sum_{(x,y)} w_k(x, y) w_i(x, y) w_j(x, y) |\nabla f(x, y)| G(x, y) \quad (3-17)$$

Finalmente el arreglo resultante se normaliza para hacerlo invariante a los efectos de contrastes o cambios en la iluminación, y para reducir los efectos de cambios no lineales en iluminación debidos a la saturación de la cámara, se limitan los valores con un umbral experimental de 0,2, de manera que los valores inferiores a 0,2 son remplazados con 0,2, y se normaliza nuevamente. La normalización se efectúa empleando la norma L_2 , como se muestra a continuación:

$$\begin{aligned} H &= (H_{11}, H_{12}, \dots, H_{nn}) \Rightarrow v = (v_1, v_2, \dots, v_m) \\ \|v\|_2 &= \sqrt{\sum_{i=1}^m v_i^2} \\ v' &= \left(\frac{v_1}{\|v\|_2}, \frac{v_2}{\|v\|_2}, \dots, \frac{v_m}{\|v\|_2} \right) \\ v'' &= (\max(v_1, 0.2), \max(v_2, 0.2), \dots, \max(v_m, 0.2)) \\ v''' &= \frac{v''}{\|v''\|_2} \end{aligned}$$

3.5. Reducción de dimensionalidad por análisis de componentes principales (KPCA)

Una vez se obtienen los vectores de características de las muestras, conviene un proceso adicional para simplificar el análisis posterior de los datos, mejorar el desempeño en la clasificación, eliminar información redundante o incluso poder obtener representaciones gráficas de los vectores. Generalmente los vectores de características resultan ser de grandes dimensiones, lo que provoca ciertas desventajas como que al aumentar las dimensiones de los vectores el volumen del espacio aumenta exponencialmente y los datos tienden a volverse dispersos, y esto afecta negativamente la clasificación, pues los datos se organizan en áreas correspondientes a grupos con características similares, y finalmente las estrategias comunes de clasificación no son eficaces. Ese efecto llamado "la maldición de la dimensión" puede ser abordado con diferentes métodos, entre estos la reducción de dimensionalidad tomando las componentes principales del grupo de datos usando el truco del *kernel* (KPCA) originalmente propuesto por Schölkopf et al. (1997).

En principio se toma el método de análisis de componentes principales (PCA), en el que básica-

mente se hace una transformación euclídea al rotar y trasladar los ejes para alcanzar la mayor variabilidad descendientemente en todas las dimensiones. En la práctica, se trazan planos de modo que las distancias de los puntos a estos sean las mínimas posibles. Las componentes principales corresponden a las primeras dimensiones del hiperplano resultante, en las que se encuentra la mayor variabilidad. Para lograr esto se diagonaliza la matriz de covarianza de los datos $\mathbf{x}_k \in \mathbf{R}^N$, con $k = 1, \dots, \ell$ definida en 3-19. Los datos están centrados en el origen, de modo que $\sum_{k=1}^{\ell} \mathbf{x}_k = 0$.

$$\mathbf{C} = \frac{1}{\ell} \sum_{j=1}^{\ell} \mathbf{x}_j \mathbf{x}_j^{\top} \quad (3-19)$$

La diagonalización, en otras palabras, es una descomposición en valores y vectores propios de la matriz \mathbf{C} , y las proyecciones ortogonales de los puntos en los eigenvectores son las componentes principales.

Ahora, suele ocurrir que la separación entre los datos no es del todo lineal y entonces es necesario hacer una transformación no lineal de los datos a un nuevo espacio de características \mathcal{F} , como se describe en (3-20). En ese nuevo espacio \mathcal{F} también es posible hacer el análisis PCA. La transformación se realiza usando *kernels*, que son funciones continuas conocidas del método de las máquinas de vectores de soporte (SVM), que además mejoran el costo computacional, porque el cálculo depende del producto interno de los vectores en el nuevo espacio, i.e $k(\mathbf{x}, \mathbf{x}') = \Phi(\mathbf{x})^{\top} \cdot \Phi(\mathbf{x}')$.

$$\Phi : \mathbf{R}^N \rightarrow \mathcal{F}, \quad \mathbf{x} \mapsto \mathbf{X} \quad (3-20)$$

Luego, si en el espacio original el análisis PCA se hacía con la descomposición de $\lambda \mathbf{x}_k \mathbf{V} = \mathbf{x}_k \mathbf{C} \mathbf{V}$, en el nuevo espacio de características, equivalentemente se hace la descomposición del sistema 3-21.

$$\lambda (\Phi(\mathbf{x}_k) \cdot \mathbf{V}) = (\Phi(\mathbf{x}_k) \cdot \bar{\mathbf{C}} \mathbf{V}), \quad \forall k = 1, \dots, \ell \quad (3-21)$$

Con $\bar{\mathbf{C}} = \frac{1}{\ell} \sum_{j=1}^{\ell} \Phi(\mathbf{x}_j) \Phi(\mathbf{x}_j)^{\top}$. Luego, el vector propio puede ser expresado como una combinación lineal de los datos transformados:

$$\mathbf{V} = \frac{1}{\ell \lambda} \sum_{i=1}^{\ell} (\Phi(\mathbf{x}_i) \cdot \mathbf{V}) \Phi(\mathbf{x}_i) \quad (3-22a)$$

$$\mathbf{V} = \sum_{i=1}^{\ell} \alpha_i \Phi(\mathbf{x}_i) \quad (3-22b)$$

Ahora, para generar el producto interno de los vectores, se multiplica a ambos lados de 3-22b

por $\Phi(\mathbf{x}_j)$, y se obtiene:

$$\mathbf{V} \cdot \Phi(\mathbf{x}_j) = \lambda \ell \alpha_j = \sum_{i=1}^{\ell} \alpha_i (\Phi(\mathbf{x}_i) \cdot \Phi(\mathbf{x}_j)) = \sum_{i=1}^{\ell} \alpha_i \mathbf{K}_j \quad (3-23)$$

Recordando que $\mathbf{K}_j := (\Phi(\mathbf{x}_i) \cdot \Phi(\mathbf{x}_j))$. Finalmente, expresándolo de forma vectorial y matricial se llega a que el problema de eigenvalores a resolver es:

$$\ell \lambda \alpha = \mathbf{K} \alpha \quad (3-24)$$

De este modo, los valores propios de \mathbf{K} son proporcionales a los valores propios de $\bar{\mathbf{C}}$ y la extracción de características se haría con los eigenvalores más grandes.

Como funciones de *kernel* pueden usarse algunos de tipo polinomial, sigmoide o, como en este caso, uno de base radial como el *kernel* gaussiano que se presenta en (3-25).

$$\mathbf{k}(\mathbf{x}, \mathbf{y}) = e^{\left(-\frac{\|\mathbf{x}-\mathbf{y}\|^2}{2\sigma^2}\right)} \quad (3-25)$$

3.6. Clasificador bayesiano normal

En la última etapa de un sistema de visión de máquina se encuentra el reconocimiento e interpretación, en la cual se utilizan diversas técnicas. Una de las más clásicas es el clasificador paramétrico supervisado basado en la teoría de decisión de Bayes formulada de forma general en (3-26).

$$P(A|B) = \frac{P(A) P(B|A)}{P(B)} \quad (3-26)$$

Aquí se toma A como hipótesis y B como la evidencia. Generalmente, se hacen simplificaciones en el modelo asumiendo que hay independencia entre las características de entrada, de forma tal que se asume que la presencia o ausencia de una característica no afecta a las otras, entonces cada característica contribuye independientemente a la probabilidad del evento A . A este caso se le llama *clasificador ingenuo*.

Ahora, en términos de clases (y_i) y características (\mathbf{X}) , (3-26) puede ser expresada como en (3-27b).

$$P(y_i|\mathbf{X}) = \frac{P(y_i) P(\mathbf{X}|y_i)}{P(\mathbf{X})} \quad (3-27a)$$

$$P(y_i | \mathbf{x}_1, \dots, \mathbf{x}_n) = \frac{P(y_i) P(\mathbf{x}_1, \dots, \mathbf{x}_n | y_i)}{P(\mathbf{x}_1, \dots, \mathbf{x}_n)} \quad (3-27b)$$

Al asumir la independencia entre las características de entrada, es posible reescribir (3-27b) usando la regla de la cadena, como se presenta en (3-28).

$$P(y_i | \mathbf{x}_1, \dots, \mathbf{x}_n) = \frac{P(y_i) \prod_{j=1}^n P(\mathbf{x}_j | y_i)}{P(\mathbf{x}_1, \dots, \mathbf{x}_n)} \quad (3-28)$$

En la práctica, el denominador de (3-28) permanece constante, y como además no depende de la clase, puede omitirse y queda que la probabilidad de una clase y_i dadas las características \mathbf{X} es proporcional a la productoria, como en (3-29).

$$P(y_i | \mathbf{x}_1, \dots, \mathbf{x}_n) \propto P(y_i) \prod_{j=1}^n P(\mathbf{x}_j | y_i) \quad (3-29)$$

Finalmente, de manera intuitiva, la clase por la que se decide el clasificador es aquella que tiene mayor probabilidad, según la expresión (3-30).

$$\hat{y} = \arg \max_{y_i} P(y_i) \prod_{j=1}^n P(\mathbf{x}_j | y_i) \quad (3-30)$$

Ahora, los parámetros del modelo, como las clases a priori y características de las distribuciones de probabilidad, se determinan sobre el conjunto de datos de entrenamiento haciendo estimaciones de máxima verosimilitud. Cuando los datos pueden tomar valores de una función continua, generalmente se asume que siguen una distribución normal, la cual se presenta su versión multivariada en (3-31).

$$\begin{aligned} p(\mathbf{x}_k) &\sim N(\boldsymbol{\mu}, \boldsymbol{\Sigma}) \\ &\sim \frac{1}{(2\pi)^{d/2} |\boldsymbol{\Sigma}|^{1/2}} \exp \left(-\frac{1}{2} (\mathbf{x}_k - \boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1} (\mathbf{x}_k - \boldsymbol{\mu}) \right) \end{aligned} \quad (3-31)$$

Para hacer la estimación de máxima verosimilitud (MLE) conviene operar (3-31) expresada en funciones monótonamente crecientes como el logaritmo natural y la suma, lo que se denomina *log-likelihood*, como se muestra en (3-32). De esta forma es más fácil maximizar la función y encontrar los parámetros de la distribución.

$$\ell(\boldsymbol{\theta}) = \sum_{k=1}^n -\frac{1}{2} (\mathbf{x}_k - \boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1} (\mathbf{x}_k - \boldsymbol{\mu}) - \frac{d}{2} \ln 2\pi - \frac{1}{2} \ln |\boldsymbol{\Sigma}| \quad (3-32)$$

Los parámetros a encontrar son $\theta_1 = \boldsymbol{\mu}$ y $\theta_2 = \boldsymbol{\Sigma}$, que al hacer la maximización de (3-32) quedan definidos en (3-33a) y (3-33b).

$$\nabla_{\theta} \equiv \begin{bmatrix} \frac{\partial}{\partial \theta_1} \\ \frac{\partial}{\partial \theta_2} \end{bmatrix} = \begin{bmatrix} \frac{\partial}{\partial \mu} \\ \frac{\partial}{\partial \Sigma} \end{bmatrix}$$

$$\nabla_{\theta} \ell(\theta) = 0$$

$$\hat{\theta}_1 = \hat{\mu} = \frac{1}{n} \sum_{k=1}^n \mathbf{x}_k \quad (3-33a)$$

$$\hat{\theta}_2 = \hat{\Sigma} = \frac{1}{n} \sum_{k=1}^n (\mathbf{x}_k - \hat{\mu})(\mathbf{x}_k - \hat{\mu})^{\top} \quad (3-33b)$$

Para conformar el clasificador normal bayesiano, en primer lugar se asume que cada clase tiene una distribución normal, de la misma forma que (3-31), y sus parámetros μ y Σ dependen de los datos que, en la etapa de entrenamiento, se ha determinado que describen la distribución normal de cada clase. Y entonces (3-31) pasa a ser la probabilidad de \mathbf{x}_j dada una clase y_i , con los parámetros (3-34a) y (3-34b).

$$\mu_i = \frac{1}{n_i} \sum_{j=1}^n z_{ij} \mathbf{x}_j \quad (3-34a)$$

$$\Sigma_i = \frac{1}{n_i} \sum_{j=1}^n z_{ij} (\mathbf{x}_j - \mu_i)(\mathbf{x}_j - \mu_i)^{\top} \quad (3-34b)$$

$$z_{ij} = \begin{cases} 1, & \text{si } \mathbf{x}_j \in y_i \\ 0, & \text{si no} \end{cases}$$

Con estas probabilidades condicionales se combina (3-31) con (3-30), y primero se definen las funciones discriminantes con *log-likelihood*.

$$\begin{aligned} g_i(\mathbf{x}) &= \ln p(\mathbf{x}|y_i) + \ln P(y_i) \\ &= -\frac{1}{2} (\mathbf{x} - \mu_i)^{\top} \Sigma_i^{-1} (\mathbf{x} - \mu_i) - \frac{d}{2} \ln 2\pi - \frac{1}{2} \ln |\Sigma_i| + \ln P(y_i) \end{aligned} \quad (3-35)$$

Finalmente, la función (3-35) se utiliza para decidir la clase a la que pertenecen las características $\mathbf{x} = x_1, \dots, x_d$ de una muestra dada. La clase de salida nuevamente es la que tenga la mayor probabilidad:

$$\hat{y} = \arg \max_{y_i} g_i(\mathbf{x}) \quad (3-36)$$

3.7. Resultados

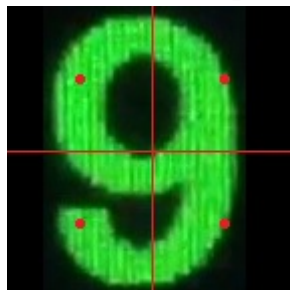
A continuación se presentarán algunos detalles en la implementación de los procesamientos presentados en las secciones anteriores y ejemplos de los resultados. Las imágenes del conjunto de entrenamiento y de prueba fueron obtenidas fotografiando la pantalla de un sonómetro marca 01dB, modelo CUBE; un ejemplo se muestra en la figura 3-2.

Figura 3-2.: Ejemplo de una fotografía de la pantalla de un sonómetro 01dB CUBE.



Para el filtro gaussiano se empleó $\sigma_x = \sigma_y = 1,5$. Luego, con el fin de simplificar el cálculo del descriptor SIFT, se escala la imagen de un dígito y se rellenan los píxeles faltantes para conformar una imagen de 32×32 . Luego, se eligen cuatro puntos clave que corresponden a los centros de cada cuadrante como se presenta en la figura 3-3.

Figura 3-3.: Puntos clave para el cálculo del descriptor SIFT de una imagen de un dígito.



En esos puntos se calcula el descriptor SIFT, no sin previamente haber umbralizado la imagen

redimensionada con el método de Otsu. De forma tal que queda un vector de características con 512 valores para cada muestra. Una breve implementación de este procesamiento se presenta en el código 3.1, que posteriormente será usado en la aplicación desarrollada para automatizar la calibración de sonómetros. Los resultados del procesamiento se pueden ver en la figura 3-4 para tres ejemplos del conjunto de muestras de entrenamiento.

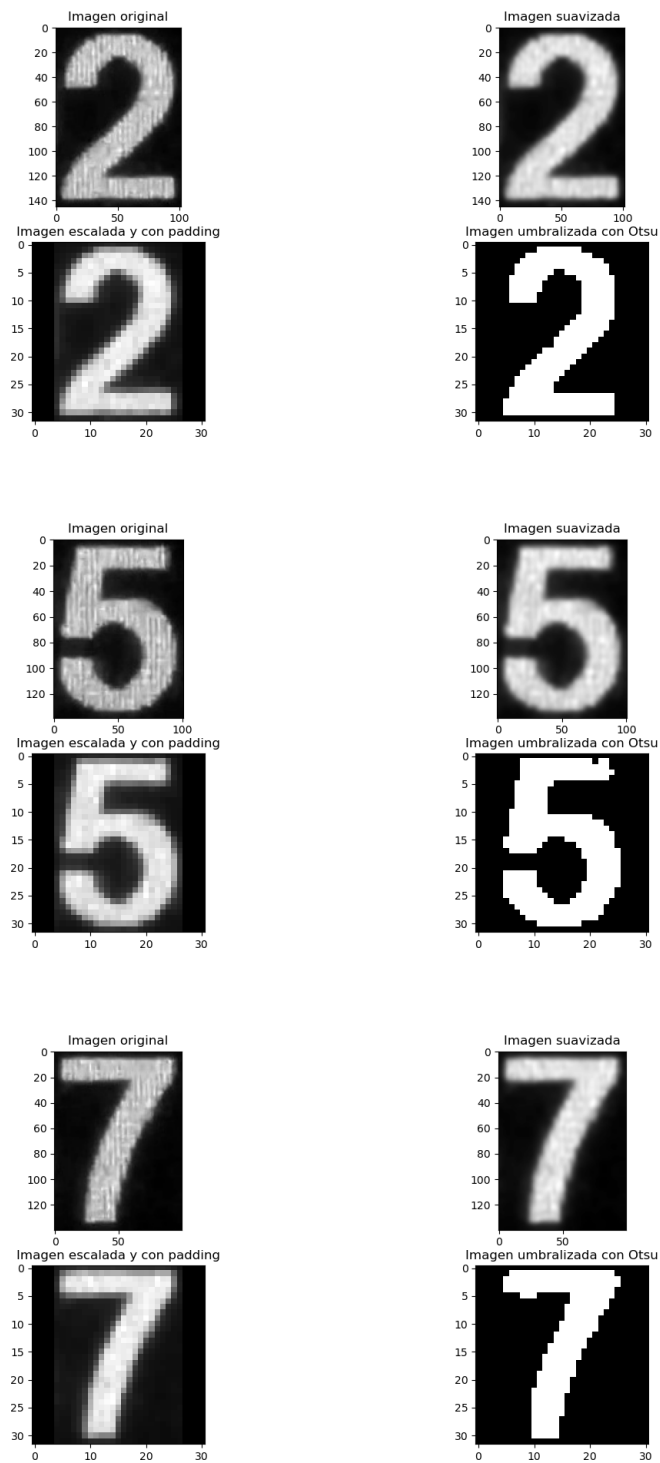
Código 3.1: Código para presentar resultados del procesamiento de 3 muestras del conjunto de imágenes entrenamiento.

```

1 import cv2 as cv
2 import matplotlib.pyplot as plt
3 import numpy as np
4
5
6 """
7 Programed by Felipe Maldonado,
8 Universidad Nacional de Colombia, Faculty of Engineering,
9 Department of Electrical and Electronic Engineering.
10 Created on 2022/03.
11 """
12 __autor__ = 'Juan Felipe Maldonado'
13 __version__ = '1.0 Beta'
14
15 path = "Images/Train"
16 img_name = ["2(0).jpg", "5(0).jpg", "7(0).jpg"] # 3 samples just for showing results of processing
17 descriptor = cv.SIFT_create(contrastThreshold=0.02, edgeThreshold=1, sigma=0.01) # Creates a SIFT descriptor
18 # Four key points for computing SIFT descriptor
19 key_points = [cv.KeyPoint(7, 7, 16), cv.KeyPoint(7, 13, 16),
20               cv.KeyPoint(13, 7, 16), cv.KeyPoint(13, 13, 16)]
21
22 X = []
23 y = []
24 for name in img_name:
25     img = cv.imread(path + "/" + name, cv.IMREAD_GRAYSCALE) # Read image in gray scale
26     img_blur = cv.GaussianBlur(img.copy(), ksize=(0, 0), sigmaX=1.5) # Gaussian filtering
27     mayor_axis = max(img_blur.shape) # The largest axis
28     # Normalize size
29     digit_scaled = cv.resize(img_blur, (0, 0), fx=32 / mayor_axis,
30                               fy=32 / mayor_axis, interpolation=cv.INTER_CUBIC)
31     # Build an image of 32 x 32 pixels with padding
32     if digit_scaled.shape[0] > digit_scaled.shape[1]:
33         digit_scaled = np.pad(digit_scaled, ((0, 0), ((32 - digit_scaled.shape[1]) // 2,
34               (32 - digit_scaled.shape[1]) // 2)), 'constant')
35     else:
36         digit_scaled = np.pad(digit_scaled, (((32 - digit_scaled.shape[0]) // 2,
37               (32 - digit_scaled.shape[0]) // 2), (0, 0)), 'constant')
38
39     img_bin = cv.threshold(digit_scaled, 0, 255, cv.THRESH_BINARY | cv.THRESH_OTSU)[1] # Otsu binarization
40     # -- SIFT DESCRIPTOR --
41     _, features = descriptor.compute(img_bin, key_points) # Computes SIFT descriptor
42     X.append(features.flatten()) # Add the features vector of the actual training sample
43     y.append(img_name[0]) # Add the class of the actual training sample
44     # -- Plots results --
45     _, axs = plt.subplots(2, 2)
46     axs[0, 0].imshow(img, cmap='gray')
47     axs[0, 0].set_title('Imagen original')
48     axs[0, 1].imshow(img_blur, cmap='gray')
49     axs[0, 1].set_title('Imagen suavizada')
50     axs[1, 0].imshow(digit_scaled, cmap='gray')
51     axs[1, 0].set_title('Imagen escalada y con padding')
52     axs[1, 1].imshow(img_bin, cmap='gray')
53     axs[1, 1].set_title('Imagen umbralizada con Otsu')
54     plt.show()
55
56 # -- Plots SIFT features by sample --
57 fig = plt.figure()
58 axes = fig.add_subplot(111)
59 axes.set_title('Vectores de características SIFT')
60 axes.scatter(range(len(X[0])), X[0], s=8, c='b', marker='s', label='Muestra de número 2')
61 axes.scatter(range(len(X[1])), X[1], s=8, c='r', marker='o', label='Muestra de número 5')
62 axes.scatter(range(len(X[2])), X[2], s=8, c='g', marker='x', label='Muestra de número 7')
63 axes.grid(visible=True)
64 plt.legend()
65 plt.show()

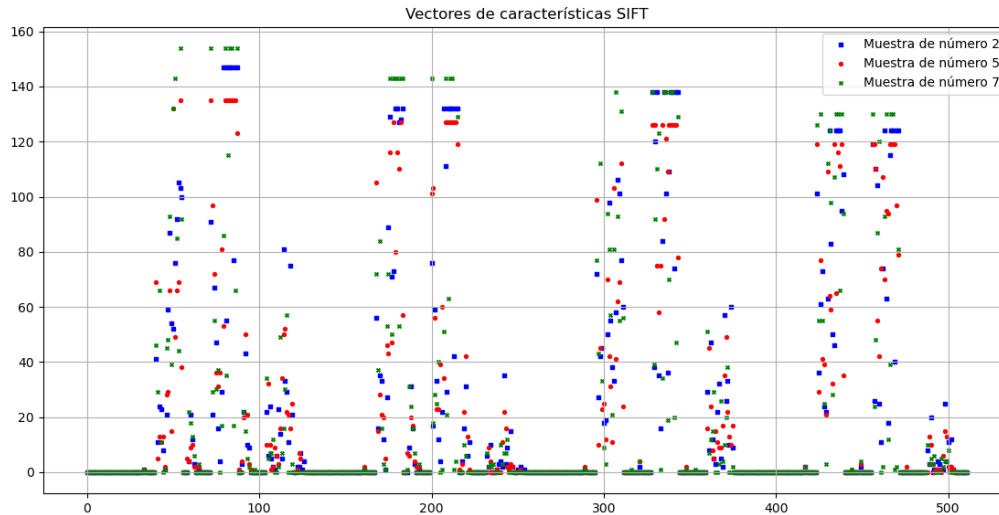
```

Figura 3-4.: Resultados de procesamiento para tres muestras del conjunto de entrenamiento.



Con las últimas líneas del código 3.1 también se grafican los vectores de características de las tres muestras. El resultado se presenta en la figura 3-5.

Figura 3-5.: Vectores de características de las tres muestras de entrenamiento procesadas.



Ahora, el código 3.1 se puede ejecutar iterativamente para leer todas las imágenes de un conjunto de entrenamiento con entre 10 y 20 muestras por cada clase. Luego, complementando con el código 3.2 se efectúa la extracción de características y se entrena el clasificador bayesiano normal. La reducción de dimensionalidad se hizo hasta un valor de $d = 16$, empíricamente seleccionado según los resultados de clasificación obtenidos.

Código 3.2: Código para realizar la extracción de características de los vectores de entrenamiento y para entrenar el clasificador.

```

1 from sklearn.decomposition import KernelPCA
2 from sklearn.naive_bayes import GaussianNB
3
4 kpcaModel = KernelPCA(n_components=16) # KPCA for dimensionality reduction
5 gnbClassifier = GaussianNB() # Gaussian naive Bayes classifier
6
7 X = np.array(X) # Convert training features vector to numpy array
8 y = np.array(y) # Convert training samples labels to numpy array
9
10 # ----- DIMENSIONALITY REDUCTION BY KERNEL PCA ----- #
11 # Fits the model and apply transformation to the features vector
12 X = kpcaModel.fit_transform(X, y)
13 # ----- FITS THE CLASSIFIER -----#
14 gnbClassifier.fit(X, y)

```

Finalmente, para un muestra dada de prueba, se efectúa el mismo procesamiento de la imagen con el código 3.1, y la extracción y clasificación se efectúa con el código 3.3.

3.8. Discusión

Los resultados obtenidos demuestran un funcionamiento satisfactorio en general, tanto del procesamiento de las imágenes como del clasificador. Aunque evidentemente es un método simplificado, resulta ser eficaz y eficiente, pues cumple con el objetivo de reconocimiento de caracteres empleando pocos recursos computacionales y cortos tiempos de entrenamiento del clasificador y del procesamiento de las imágenes, lo que permite que el método sea empleado en ordenadores con baja capacidad de procesamiento y en aplicaciones de funcionamiento en tiempo real, como se pretende en este proyecto. Además, aunque el clasificador bayesiano es de las técnicas más sencillas de aprendizaje de máquina, tiene un desempeño aceptable en muchas aplicaciones reales y no requiere una base de datos de entrenamiento demasiado extensa (scikit Developers, 2022).

Particularmente, cada una de las etapas de procesamiento resulta ser crucial en el resultado. Por ejemplo la etapa de suavizado permite eliminar variaciones de intensidad de alta frecuencia como las irregularidades en el relleno de los dígitos debidas a la construcción de la pantalla del sonómetro, como en pantallas led, por ejemplo. Sin este suavizado es posible que en la umbralización no se obtengan segmentos homogéneos que representen correctamente un dígito. La binarización de la imagen, buscando el umbral con el método de Otsu, garantiza que se haga una discriminación correcta entre el fondo de la pantalla y el dígito, en función de la distribución de las intensidades de píxeles, esto aporta confiabilidad en la segmentación independientemente de la forma y color de los números. Y el descriptor SIFT es bien conocido por su robustez frente a transformaciones afines o cambios en el punto de vista 3D, lo que contribuye a asegurar la eficacia en la medición de características aún cuando hay variaciones en las condiciones normales de adquisición de las imágenes; además, que esté basado en la dirección de los gradientes, lo hace adecuado para cuantizar de algún modo las formas de los contornos de los dígitos.

No obstante, a pesar de la eficiencia y simplicidad del clasificador bayesiano normal, lo cierto es que el resultado de clasificación es bastante sensible si se obtienen resultados erróneos en el procesamiento anterior. Por ejemplo, por efectos de bajo contraste o desenfoques que deforman los segmentos, podrían haber confusiones en la clasificación, principalmente en clases que son muy similares en sus características, como entre el 6 y el 8 o entre el 9 y el 8, pues al deformarse el segmento a causa de esos ruidos, el 6 o el 9 cierran el trazo faltante y se asemejan a un 8. Sin embargo, en los ensayos del sistema de reconocimiento se pudo comprobar que esto ocurre cuando la posición o configuración de la cámara no es la adecuada y los efectos del desenfoque o la iluminación son más pronunciados. Cuando las condiciones de captura de la imagen son controladas, la eficacia del clasificador es de un 100 %, como se puede notar en la matriz de confusión de la figura 3-6.

4. Implementación de los procedimientos de calibración

4.1. Automatización de la calibración periódica de calibradores acústicos

4.1.1. GRAFCET descriptivo del proceso

4.1.2. Implementación en Python

Arquitectura de software

4.2. Automatización de la calibración periódica de sonómetros

4.2.1. Implementación en Python

Arquitectura de software

5. Modelamiento de incertidumbre de medición en la calibración periódica de sonómetros

5.1. Cadenas de Markov de tiempo discreto

5.1.1. Matriz de probabilidades de transición

5.1.2. Valor esperado

5.1.3. Probabilidad estacionaria

5.2. Incertidumbre de medición

Definición matemática del mesurando

5.2.1. Incertidumbre tipo A

5.2.2. Incertidumbre tipo B

5.2.3. Incertidumbre típica combinada

5.2.4. Grados efectivos de libertad, factor de cobertura e incertidumbre expandida

5.3. Algoritmo para creación de estados y tasas de transición de la cadena de Markov

5.3.1. Implementación de la cadena de Markov en Python

6. Conclusiones y recomendaciones

6.1. Conclusiones

Las conclusiones constituyen un capítulo independiente y presentan, en forma lógica, los resultados de la tesis o trabajo de investigación. Las conclusiones deben ser la respuesta a los objetivos o propósitos planteados. Se deben titular con la palabra conclusiones en el mismo formato de los títulos de los capítulos anteriores (Títulos primer nivel), precedida por el numeral correspondiente (según la presente plantilla).

6.2. Recomendaciones

Se presentan como una serie de aspectos que se podrían realizar en un futuro para emprender investigaciones similares o fortalecer la investigación realizada. Deben contemplar las perspectivas de la investigación, las cuales son sugerencias, proyecciones o alternativas que se presentan para modificar, cambiar o incidir sobre una situación específica o una problemática encontrada. Pueden presentarse como un texto con características argumentativas, resultado de una reflexión acerca de la tesis o trabajo de investigación.

A. Anexo: Certificado de calibración de calibrador acústico

B. Anexo: Certificado de calibración de sonómetro

Bibliografía

Brüel & Kjær (2016). *Instruction Manual, Handheld Analyzer Types 2250, 2250L and 2270*. Brüel & Kjær Sound & Vibration Measurement A/S, Nærum, Denmark.

Brüel & Kjær (2021). What is a Sound Level Meter? <https://www.bksv.com/en/knowledge/blog/sound/what-is-a-sound-level-meter>. En línea: consultada en marzo del 2022.

IEC technical committee 29: Electroacoustics (2017). *Electroacoustics – sound calibrators*. Standard IEC 60942:2017, International Electrotechnical Commission, Geneva, Switzerland.

Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110.

Otsu, N. (1979). A Threshold Selection Method from Gray-Level Histograms. *IEEE Trans Syst Man Cybern*, SMC-9(1):62–66.

Richard, S. (2011). *Algorithms and Applications: Computer vision*. Springer, London, 2 edition.

Schölkopf, B., Smola, A., and Müller, K. R. (1997). *Kernel principal component analysis*, volume 1327. Springer, Alemania.

Scikit-learn developers (2022). 1.9. Naive Bayes. https://scikit-learn.org/stable/modules/naive_bayes.html#gaussian-naive-bayes. En línea: consultada en marzo del 2022.