

Capítulo 5

Experimentos y resultados

En este capítulo se va a probar el funcionamiento de los algoritmos diseñados para las aplicaciones que se han desarrollado. Para ello se analizarán distintas imágenes y vídeos en diversas condiciones, y con varias posibilidades de configuración.

Se comentarán los resultados obtenidos y se sacarán conclusiones respecto al funcionamiento de los algoritmos y de los parámetros de configuración.

Con las dos aplicaciones desarrolladas se pueden analizar gran cantidad de imágenes y vídeos, ese es su objetivo. En este capítulo se recogen algunos experimentos, para ilustrar la conveniencia de la elección de ciertos valores en los parámetros de configuración, pero las posibilidades son mucho más amplias.

5.1. Sustracción del fondo de la imagen

Tanto SparksSolver como SparksAnalyzer sustraen una imagen de fondo a la imagen que se está analizando. SparksSolver permite ver el resultado en pantalla.

En la figura 5.1 se muestra el funcionamiento de la operación. En este caso el resultado es satisfactorio, sólo quedan las chispas y un leve ruido donde antes estaba la muela. El motivo de este buen funcionamiento es que la cámara ha permanecido inmóvil entre ambas imágenes.

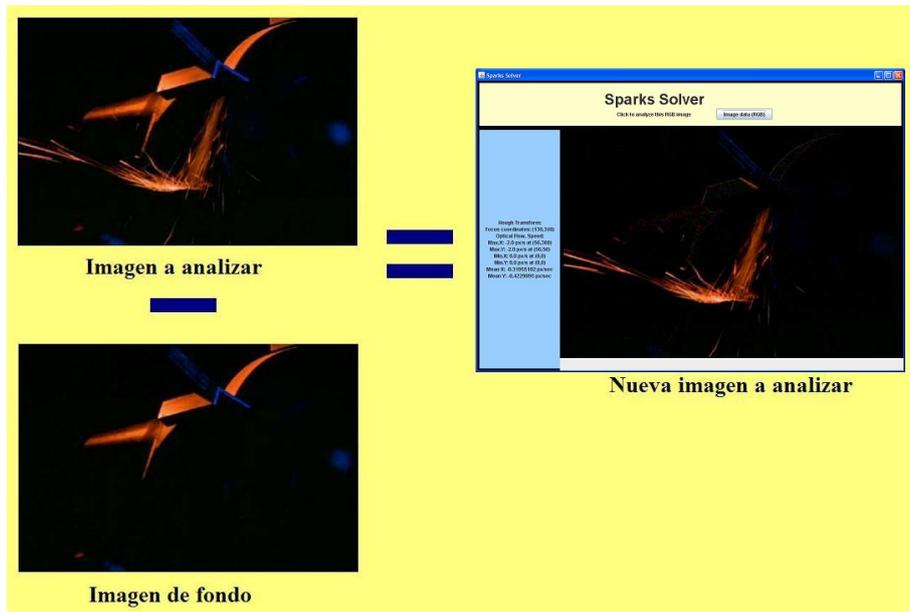


Figura 5.1: Sustracción correcta.

Si se aplica el algoritmo a las imágenes *Rolls.jpg* y *RollsBG.jpg*, se obtiene lo que aparece en la figura 5.2. La imagen de fondo contenía chispas, y además hay un pequeño desplazamiento de la cámara entre las dos imágenes, ya que se obtuvieron con la cámara en mano. Esto provoca que la sustracción no sea totalmente efectiva, ya que queda ruido en la imagen resultante. Esto a su vez influye de forma significativa en el funcionamiento de los algoritmos que se le aplican. En particular, el del flujo óptico, ya que hay más diferencia entre las imágenes que la que hay entre las escenas reales. El gráfico del flujo óptico aparece en la figura 5.3, y se aprecia mucho más movimiento en él del que realmente existe.

Pero este ejemplo ilustra un resultado importante: se pueden eliminar fuentes de iluminación estáticas que aparezcan en la imagen. El foco de la parte superior izquierda desaparece. De este modo no afectaría a la discriminación de las chispas del resto de la imagen, que se lleva a cabo por intensidad luminosa.

Otro ejemplo de sustracción correcta por la adquisición de imágenes en condiciones adecuadas se muestra en la figura 5.4. Las imágenes utilizadas son *heavygrindingBG.jpg* y *heavygrinding480000000.jpg*, ambas proporcionadas con SparksSolver.

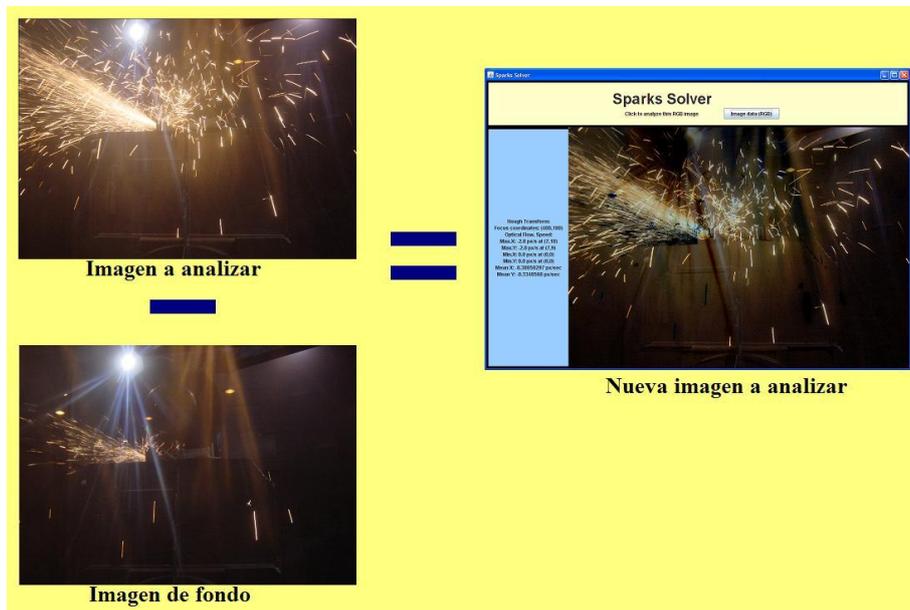


Figura 5.2: Sustracción con desplazamiento de la cámara.

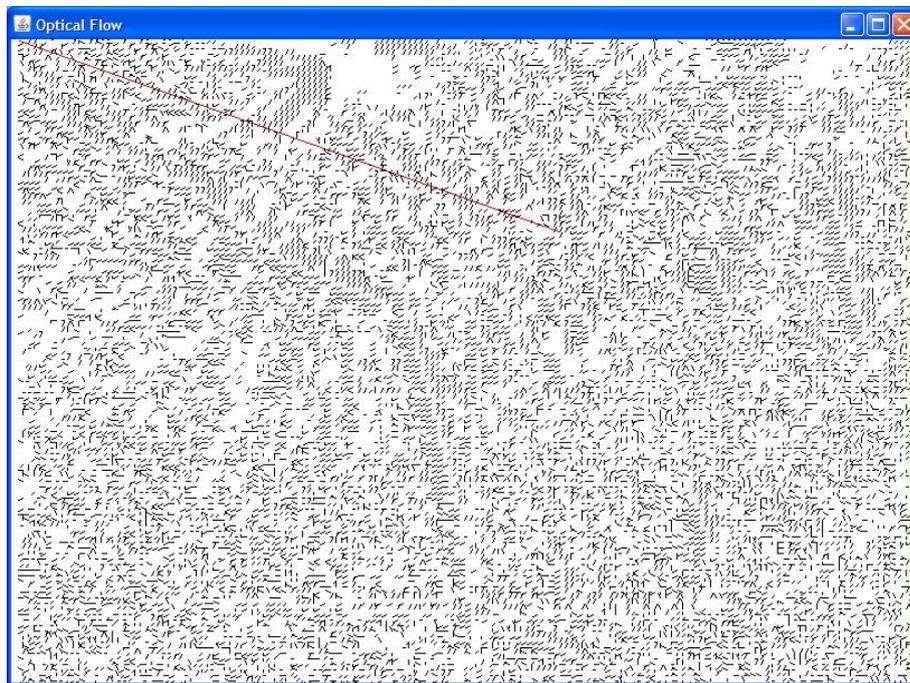


Figura 5.3: Flujo óptico tras una sustracción inadecuada.

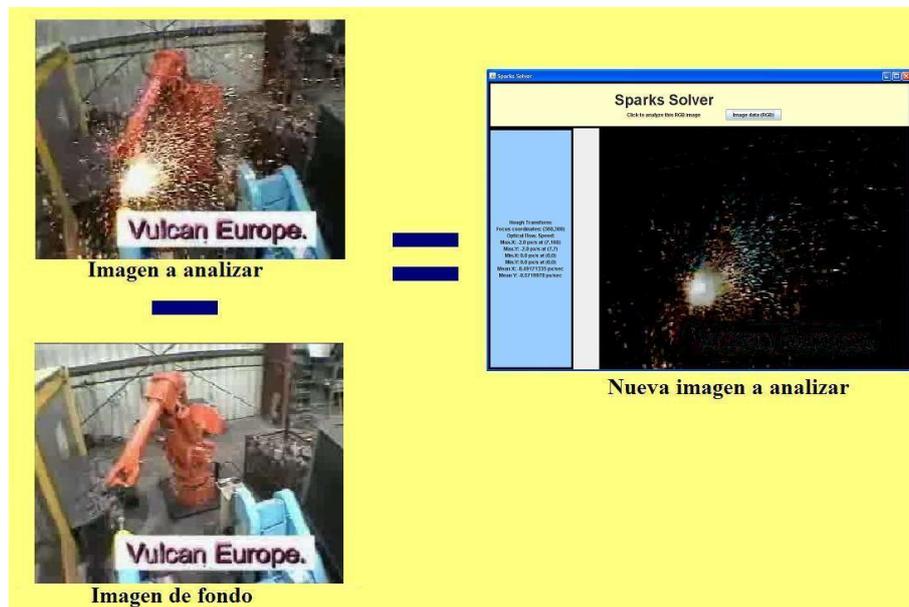


Figura 5.4: Sustracción correcta.

Este método tiene ventajas e inconvenientes. Por una parte facilita la aplicación de algoritmos complejos y reduce su coste computacional. Además elimina elementos que no son de interés, o que estorban al análisis. A cambio exige ciertas condiciones en la imagen de fondo. Ésta debe ser lo más parecida posible a una imagen de trabajo, sobre todo en lo que se refiere a la posición de la cámara y del brazo robótico. Esto limita la libertad de actuación, con lo cual hay que llegar a un compromiso entre calidad del análisis y libertad de movimientos.

Por otra parte, para facilitar la labor de estos programas de visión, convendría que el lugar de trabajo fuera lo más uniforme posible en cuanto a color. Lo ideal sería que tanto el lugar como el robot fueran negros. De este modo los resultados serían mejores ya que al reducir los elementos que no tienen interés, la sustracción los eliminaría casi totalmente.

5.2. Cómputo del área de chispas

Para evaluar el tamaño de las chispas de una imagen, se ha optado por umbralizar las imágenes en intensidad. Así, se considera que pertenecen a chispas aquellos píxeles con más de un cierto nivel de luminancia. Dadas las características del proceso de rectificado, cuando hay chispas, éstas se corresponden con los niveles más altos de intensidad de la imagen, de manera que para segmentarlas del resto de la imagen basta con escoger adecuadamente el valor del umbral.

Las dos aplicaciones desarrolladas permiten al usuario seleccionar un valor de umbral para computar el área de las chispas. Cuando no se sabe qué valor elegir, el primer paso es realizar varios análisis probando distintos valores con SparksSolver. A continuación se podrán analizar las numerosas imágenes que componen un vídeo configurando en SparksAnalyzer el valor obtenido anteriormente.

En el caso de las imágenes de la figura 5.4, se han probado varios valores: 50, 100, 150 y 200. Cabe recordar en este momento que en Java los píxeles de las imágenes con bandas RGB son números enteros compuestos por ternas de enteros entre 0 y 255. Una imagen en blanco y negro tiene una sola banda con valores de intensidad comprendidos entre 0 y 255. Los resultados obtenidos con los valores de umbral probados se muestran en la figura 5.5. Parece entonces que el valor más adecuado se encuentra entre 50 y 100. Las áreas calculadas para cada zona aparecen en el fichero de salida:

Umbral=50,	Umbral=150,
ROI size: 23773 px 5.162432% total.	ROI size: 4733 px 1.0277959% total.
Umbral=100,	Umbral=200,
ROI size: 11442 px 2.4846907% total.	ROI size: 855 px 0.18566775% total.

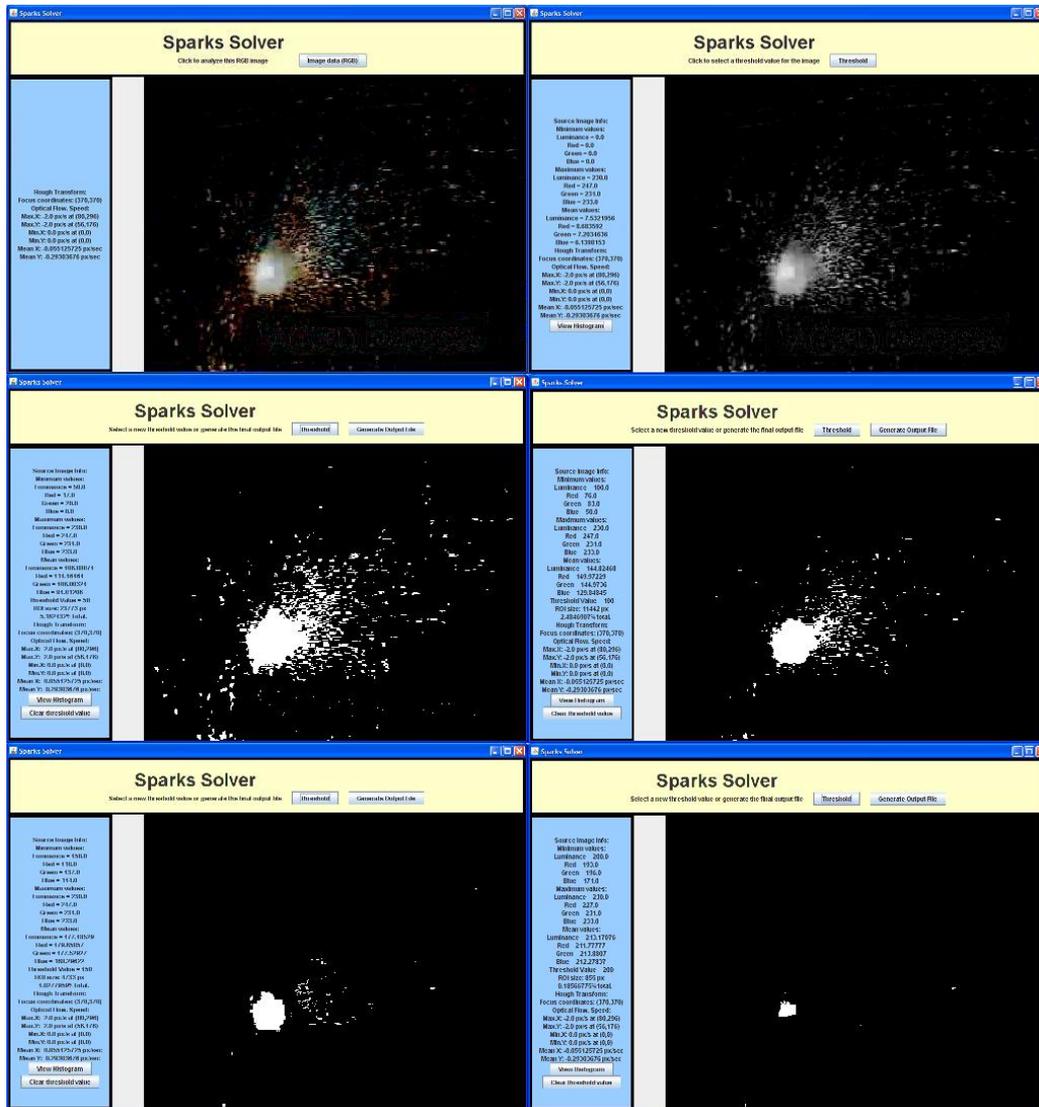


Figura 5.5: Umbralización con distintos niveles de intensidad. De izquierda a derecha y de arriba a abajo: Imagen a analizar, imagen en blanco y negro, umbral 50, umbral 100, umbral 150, umbral 200.

5.3. Transformada de Hough

Como se explicó en el Capítulo 2, el algoritmo de la transformada de Hough consiste básicamente en los siguientes pasos:

- Convertir la imagen en color a blanco y negro.
- Detectar los bordes de la imagen, con un operador como el de Sobel.
- Binarizar la imagen de bordes.
- Aplicar la transformada de Hough.
- Reducir las nubes de puntos que corresponden a una misma línea.
- Representar las líneas obtenidas.

El algoritmo implementado en las aplicaciones desarrolladas introduce variaciones para mejorar su funcionamiento con imágenes como las que nos ocupan. Estas mejoras se describen a continuación.

5.3.1. Mejora del algoritmo clásico

La primera mejora es que la binarización se realiza con histéresis y con umbrales dinámicos. Se aplica en primer lugar un umbral elevado, y a los vecinos de los puntos que lo superen, un umbral de menor valor. Los valores de los umbrales no son fijos. Para mayor versatilidad del algoritmo, se asignan de forma dinámica con una tabla, según el histograma de la imagen.

De esta forma, al aplicar primero un umbral elevado, se reduce la presencia de puntos aislados, reduciendo por consiguiente el ruido.

Además se ha modificado el primer paso del algoritmo. En lugar de realizar la detección de bordes a la imagen en blanco y negro, se ha realizado sobre la banda azul de la imagen en color. Este cambio se ha basado en el estudio de las características de las imágenes con chispas. Se ha descubierto que la banda azul contiene menos elementos que no sean chispas. Las bandas roja y verde presentan el ruido que queda tras la sustracción de la imagen de fondo a la imagen a analizar. La comparativa se muestra en la figura 5.6. Como se ha señalado en la sección 5.1, los elementos que quedan en la imagen

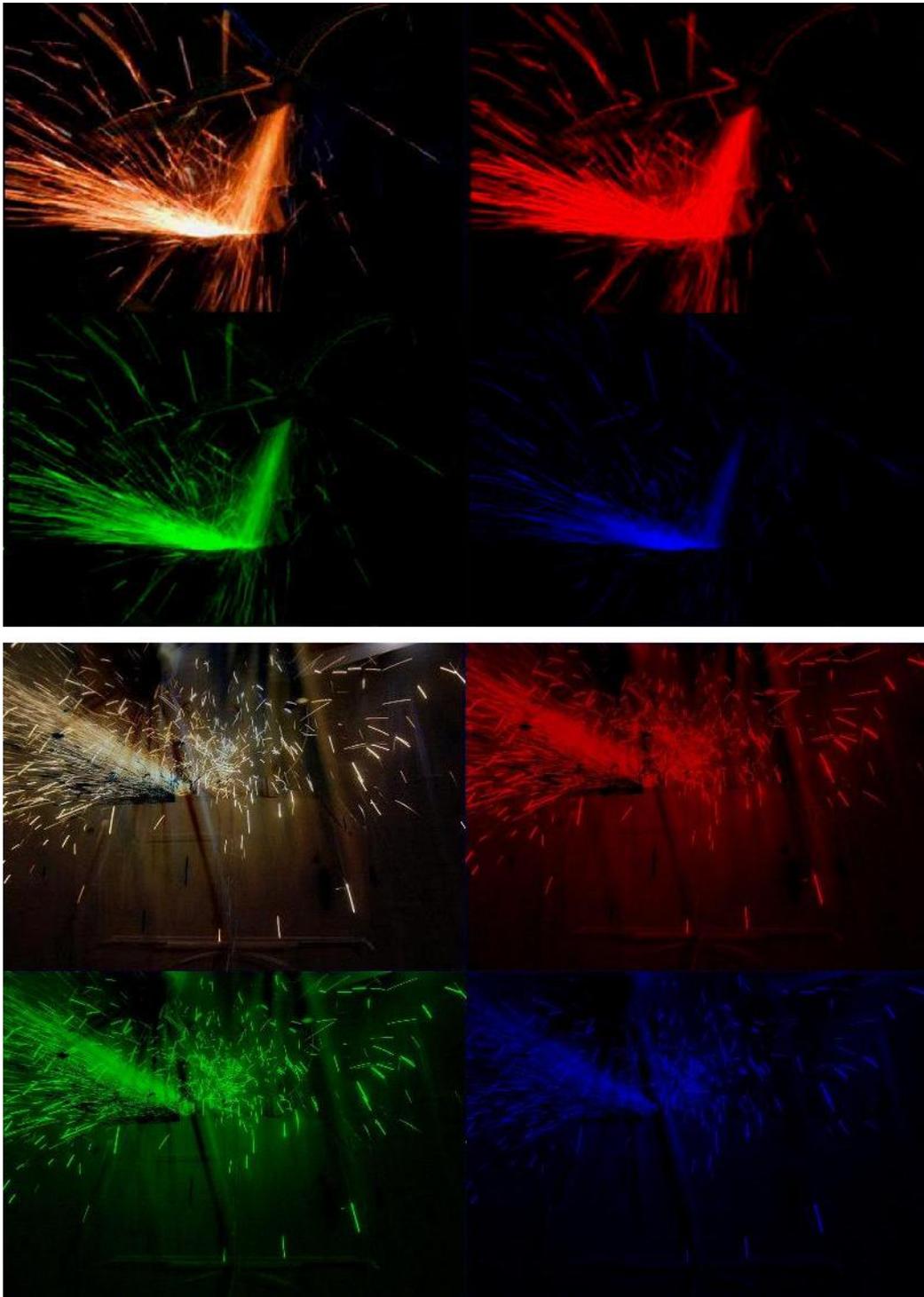


Figura 5.6: Comparación entre el contenido de las bandas de la imagen en color, con sustracción de buena calidad (arriba) y mala (abajo).

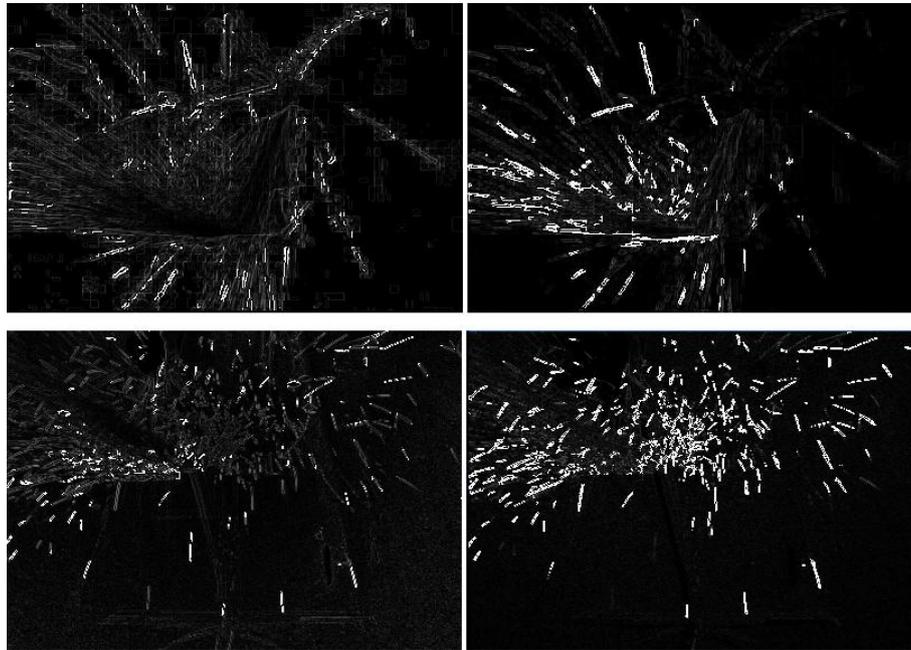


Figura 5.7: Comparación entre la detección de bordes de una imagen en blanco y negro (izquierda) frente a la banda azul de una imagen en color (derecha).

tras la sustracción del fondo son prácticamente inapreciables a nuestros ojos, pero provocan cierta cantidad de ruido en la imagen. Podría pensarse que este ruido no afecta al funcionamiento de la detección de bordes mediante el operador de Sobel, pero se hecho se detectan elementos del fondo. Esto es más significativo cuanto peor haya sido la sustracción del fondo. En la figuras 5.7 puede apreciarse la diferencia en la detección de bordes si se utiliza la banda de luminancia o la banda azul, para las imágenes anteriores.

Al eliminar ruido y elementos que no son de interés, se logra una gran mejora la aplicación del algoritmo de la transformada de Hough para la detección de la trayectoria de las chispas.

5.3.2. Número de líneas detectadas

En el algoritmo de la transformada de Hough es posible configurar el número de líneas que se va a detectar en la imagen. Se muestran las líneas con mayor número de puntos. El número de líneas se debe elegir lo suficientemente grande como para obtener las trayectorias significativas, pero no tanto como para detectar líneas que no se corresponden a chispas.

En la figura 5.8 se muestran los resultados de varios análisis con Sparks-Solver, con tamaño de la zona del foco 20px·20px y radios de los clusters para una misma línea igual a 10. Se ha probado a detectar 1, 5, 10 y 15 líneas. De los resultados se deduce que cinco líneas son insuficientes, y 15 son demasiadas ya que aparecen líneas que no se corresponden a trayectorias de las chispas. Con 10 líneas se identifican prácticamente todas las trayectorias, por tanto para esta imagen se deberían solicitar entre 5 y 10 líneas.

En la figura 5.9 se ha analizado otra imagen, con tamaño de la zona del foco 20px·20px y radios de los clusters para una misma línea igual a 10. Se ha probado a detectar 1, 5, 10, 15 y 20 líneas. Como puede verse, una línea es insuficiente, y 15 son demasiadas ya que aparecen líneas que no se corresponden a trayectorias de las chispas. Con 10 líneas quedan algunas trayectorias por identificar, con lo cual lo ideal en estas condiciones sería solicitar un número entre 10 y 15.

5.3.3. Radio de los clusters

Además las líneas detectadas dependen del radio de los clusters configurado. En la figura 5.10 se muestra una comparativa entre los resultados análisis, con tamaño de la zona del foco 20px·20px, 10 y 15 líneas, y radios de los clusters para una misma línea 5, 10 y 20. Puede apreciarse que con radios 5 y 20, el foco no se detecta correctamente, mientras que con 10 el resultado es satisfactorio. En cuanto a las trayectorias detectadas, con 10 líneas y radio 10, el resultado es mejor que con el mismo radio y 15 líneas. Con 15 líneas y radio 10 se obtiene prácticamente el mismo resultado que con 10 líneas y radio 20. En este caso por tanto, el mejor resultado se obtiene con 10 líneas y radio 10. Y este sería el valor de los parámetros que se debería introducir en los análisis con SparksAnalyzer.

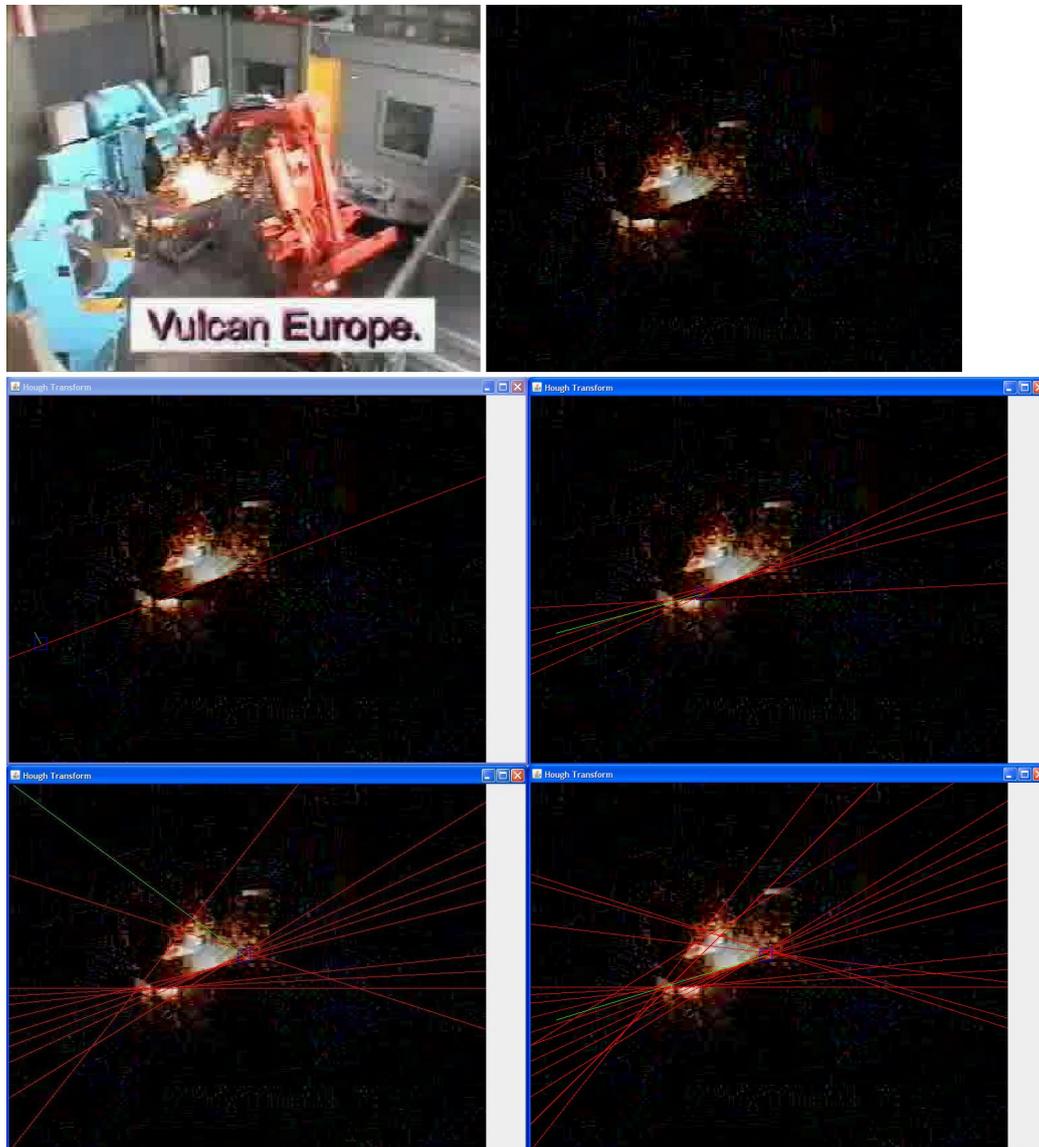


Figura 5.8: Comparativa entre las trayectorias detectadas según el número solicitado. De izquierda a derecha y de arriba a abajo: imagen original, imagen sin fondo, y 1, 5, 10 y 15 líneas detectadas.

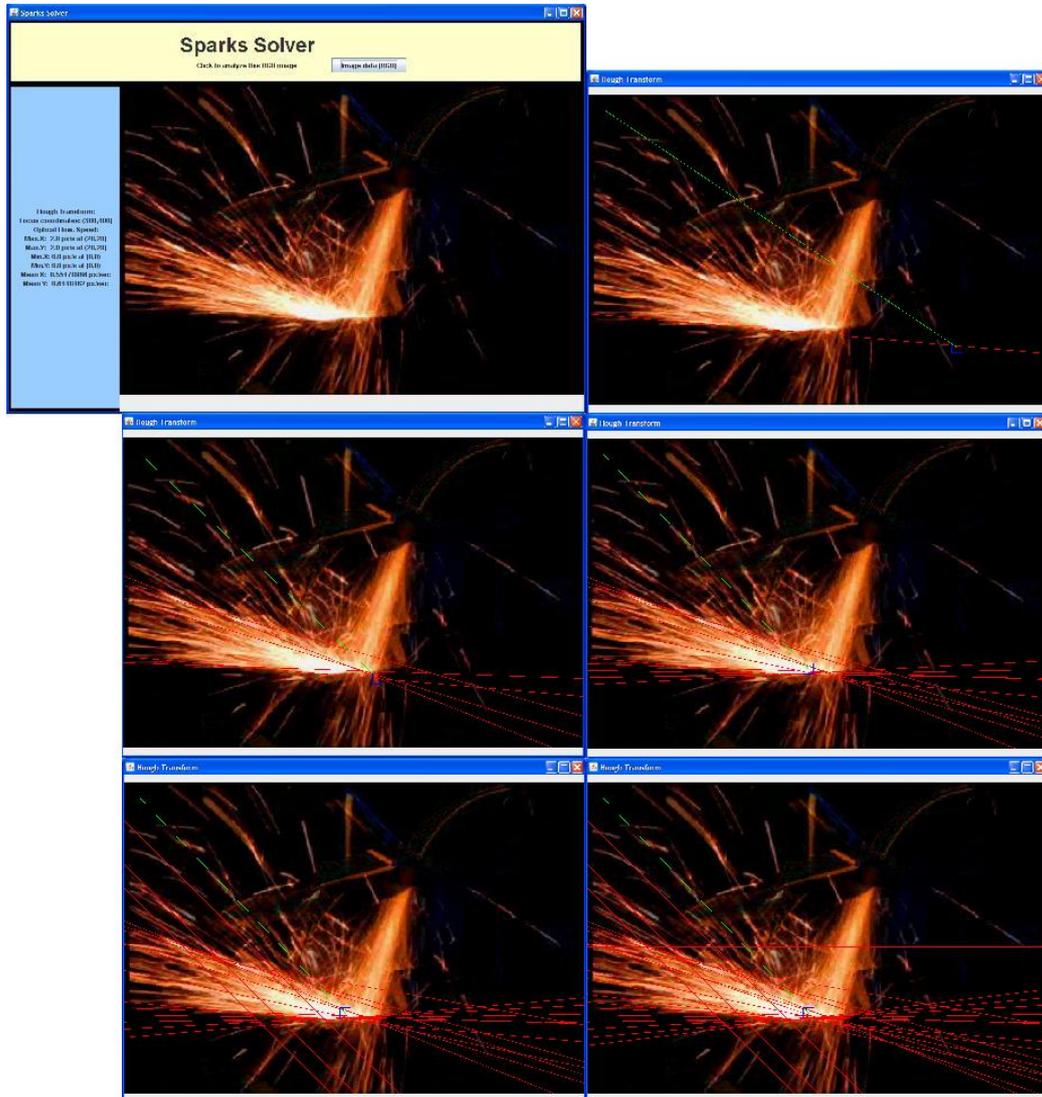


Figura 5.9: Comparativa entre las trayectorias detectadas según el número solicitado. De izquierda a derecha y de arriba a abajo: imagen original, y 1, 5, 10, 15 y 20 líneas detectadas.

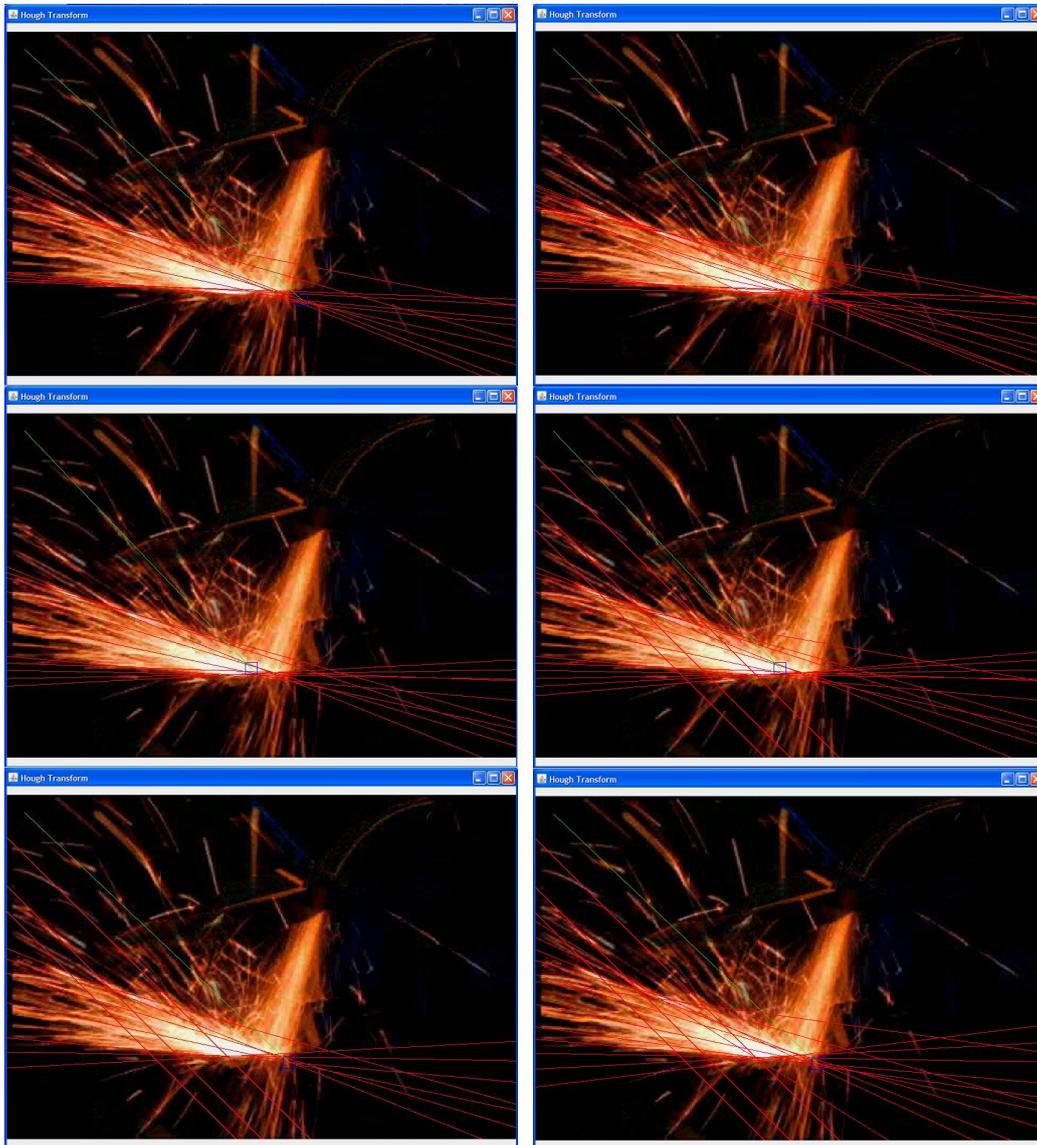


Figura 5.10: Comparativa entre las de trayectorias detectadas según el radio de las nubes de puntos que corresponden a una sola línea. Izquierda: 10 líneas y radios 5, 10 y 20. Derecha: 15 líneas y radios 5, 10, 20.

5.4. Identificación del foco de las chispas

5.4.1. Precisión

La identificación del foco de las chispas se lleva a cabo a partir de los resultados de la transformada de Hough. Según las trayectorias obtenidas, se identifica la zona de la imagen por la que pasan más líneas, y esa se identifica como la región del foco. Por todo esto, el resultado depende de la calidad del funcionamiento de la transformada de Hough. Si se detectan líneas que no se corresponden con trayectorias de chispas, se puede falsear el resultado de la identificación.

En la figura 5.10 se aprecia la dependencia de la identificación del foco con los resultados de la transformada de Hough. Se muestran análisis con tamaño de la zona del foco 20px·20px, 10 y 15 líneas, y radios de los clusters para una misma línea 5, 10 y 20. Puede apreciarse que con radios 5 y 20, el foco no se ha identificado correctamente debido a que las trayectorias detectadas confluyen en otra zona.

5.4.2. Tamaño de la región del foco

Además de la dependencia del número de líneas detectadas y del radio de las nubes de puntos que corresponden a una misma línea, existe otro parámetro que influye en la identificación: el tamaño de la región del foco.

En primer lugar, según el tamaño de la región y su localización incidirán en ella unas determinadas líneas. Con lo cual la zona con mayor número de puntos de líneas puede variar de una ejecución a otra según el valor del parámetro ‘Focus zone area in px·px’. En la figura 5.11 se muestran los resultados de identificaciones con tamaños 10x10, 20x20, 30x30 y 40x40. Los resultados son mejores con los tamaños 10x10 y 20x20.

En segundo lugar hay que tener en cuenta también el verdadero tamaño de la zona del foco en la imagen. Es decir, para una imagen tomada desde lejos, en la que la zona de trabajo tenga una dimensión de 100x100 píxeles, habrá que elegir un tamaño de la región menor que para otra de 400x400. Así los resultados serán más precisos.

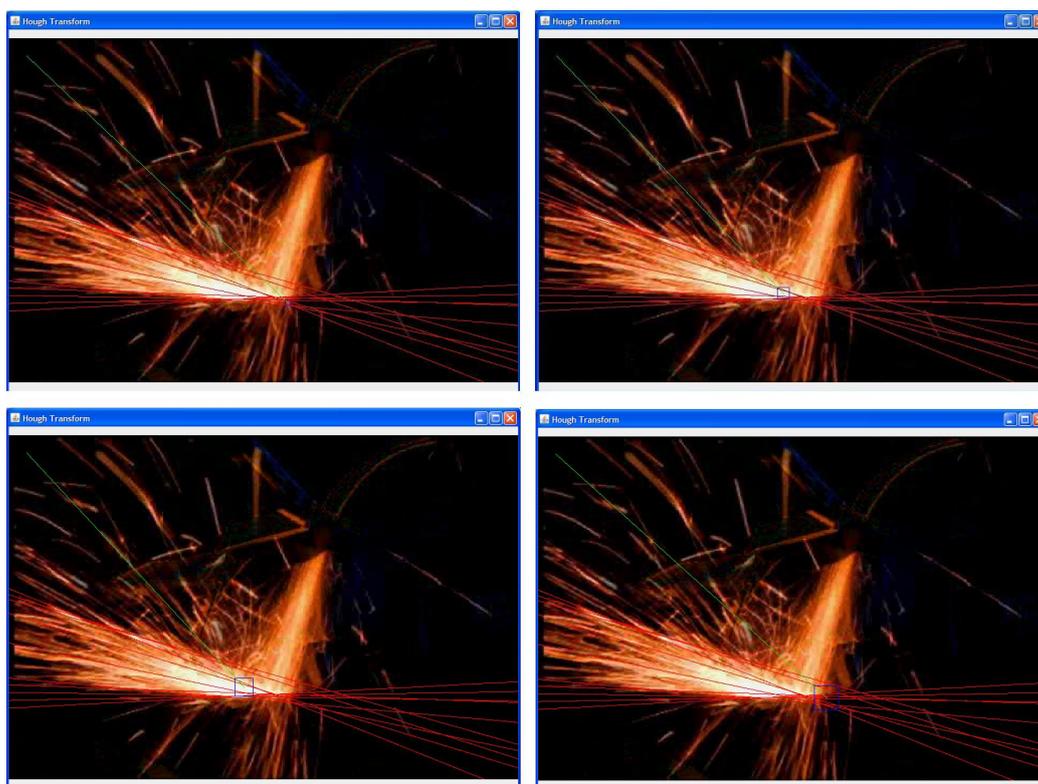


Figura 5.11: Identificación de la región del foco según el tamaño de la región. Tamaños: 10x10, 20x20, 30x30 y 40x40 píxeles.

5.5. Flujo óptico

El funcionamiento del algoritmo para el cálculo del flujo óptico es igual en las dos aplicaciones desarrolladas. Sin embargo varía la presentación de los resultados. En SparksSolver se presenta en una ventana el gráfico resultante del cálculo de las velocidades de los puntos de la imagen y se integran los resultados con los de la transformada de Hough. En SparksAnalyzer, se utiliza la información obtenida para comprobar la corrección del cálculo del foco de las chispas, pero no se presenta el gráfico para disminuir el coste computacional.

En ambas aplicaciones se puede configurar la resolución del análisis escalando o no las imágenes analizadas. Esto se selecciona con el parámetro 'Resolution'. Se pueden multiplicar las dimensiones de la imagen por 1, 0.5, 0.25 y 0.125. De nuevo la idea es ajustar en SparksSolver el menor valor para el cual los resultados son aceptables, e introducir este valor en la configuración de SparksAnalyzer.

En la figura 5.12 se muestran los resultados del algoritmo del flujo óptico para valores de 'Resolution' 0.125, 0.25, 0.50 y 1, respectivamente. La línea roja une el punto de máxima velocidad con el foco de las chispas. Como puede observarse, el punto de máxima velocidad apenas varía con el cambio de resolución, con lo que se consigue disminuir apreciablemente el coste computacional sin sacrificar apenas precisión.

5.5.1. Parámetros

En cuanto a los parámetros de configuración, se puede modificar el tamaño de la ventana de la imagen de la que se busca coincidencia, y el desplazamiento permitido en la búsqueda.

Se van a ilustrar varios experimentos variando estos parámetros. En la figura 5.13 se muestran los gráficos obtenidos con resolución 0.5, tamaños de ventana de 2x2, 4x4 y 6x6, de izquierda a derecha; y con desplazamientos 2, 4 y 6 de arriba a abajo.

La velocidad de ejecución es menor cuanto menor es el tamaño de la ventana, y menor el desplazamiento permitido. Pero las diferencias entre los gráficos apenas son apreciables.

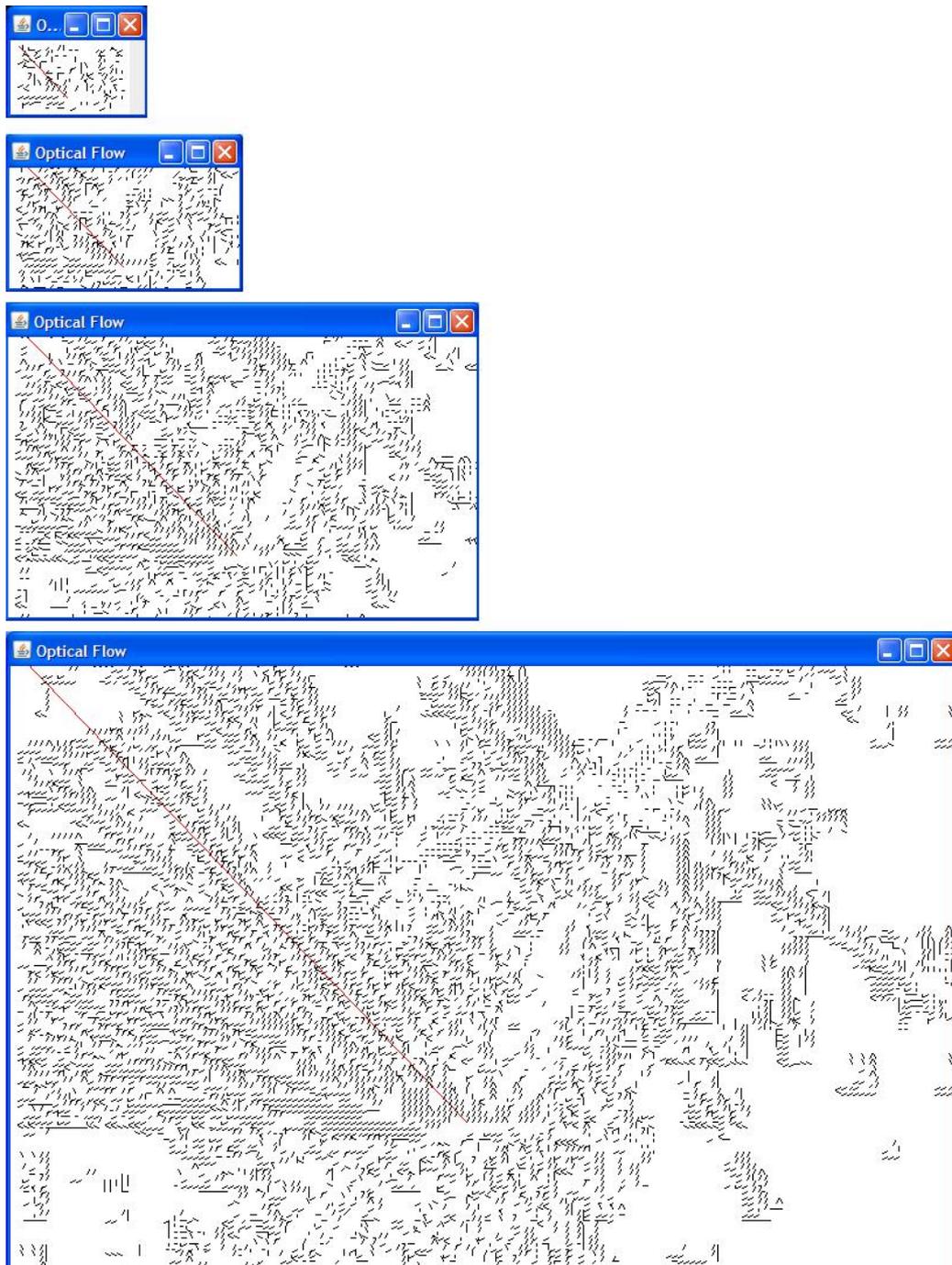


Figura 5.12: Gráficos de flujo óptico según resolución.



Figura 5.13: Variación de los parámetros del flujo óptico. De izquierda a derecha, tamaños de ventana de 2x2, 4x4 y 6x6 píxeles. De arriba a abajo, desplazamientos 2, 4 y 6 píxeles.

5.5.2. Consideraciones

La calidad de las imágenes repercute en el funcionamiento del algoritmo. En la figura 5.14 aparece un ejemplo. La imagen es de baja resolución, y aunque la sustracción del fondo es satisfactoria, existen diferencias apreciables entre la imagen de fondo y la que se está analizando, porque los píxeles son ‘demasiado grandes’ y se tiene ruido en la sustracción. Esto provoca que aparezca movimiento donde realmente no lo hay.

5.6. Integración del flujo óptico y la transformada de Hough

5.6.1. Funcionamiento

Los datos procedentes de los algoritmos del cálculo de la transformada de Hough y del flujo óptico se integran para determinar si la detección del foco de las chispas se ha llevado a cabo correctamente.

Para ello se calcula y se traza una línea entre el punto de mayor velocidad, procedente del flujo óptico, con el centro del foco detectado. Esa línea se representa gráficamente sobre la imagen con las trayectorias detectadas tras la transformada de Hough, para dar una idea visual de dónde se encuentra el punto de máxima velocidad.

En la figura 5.15 se muestran dos resultados obtenidos. En ambos la línea trazada está en un extremo del chorro de chispas.

5.6.2. Discriminación de focos erróneos

Para estimar si la identificación de la región del foco se ha llevado a cabo correctamente, se calculan las coordenadas polares de la línea que une el punto de máxima velocidad con el foco, y se comparan con las de las trayectorias obtenidas en el algoritmo de Hough. Si el valor de theta calculado no se encuentra en el rango delimitado por los valores extremos de theta de las líneas - con tolerancia dada por el parámetro de configuración ‘distance’-, el programa imprime un aviso, ya que no se tiene la certeza de haber identificado correctamente el foco de las chispas. Esto no quiere decir que la identificación no sea correcta, sino que no se tiene la certeza de que lo sea.

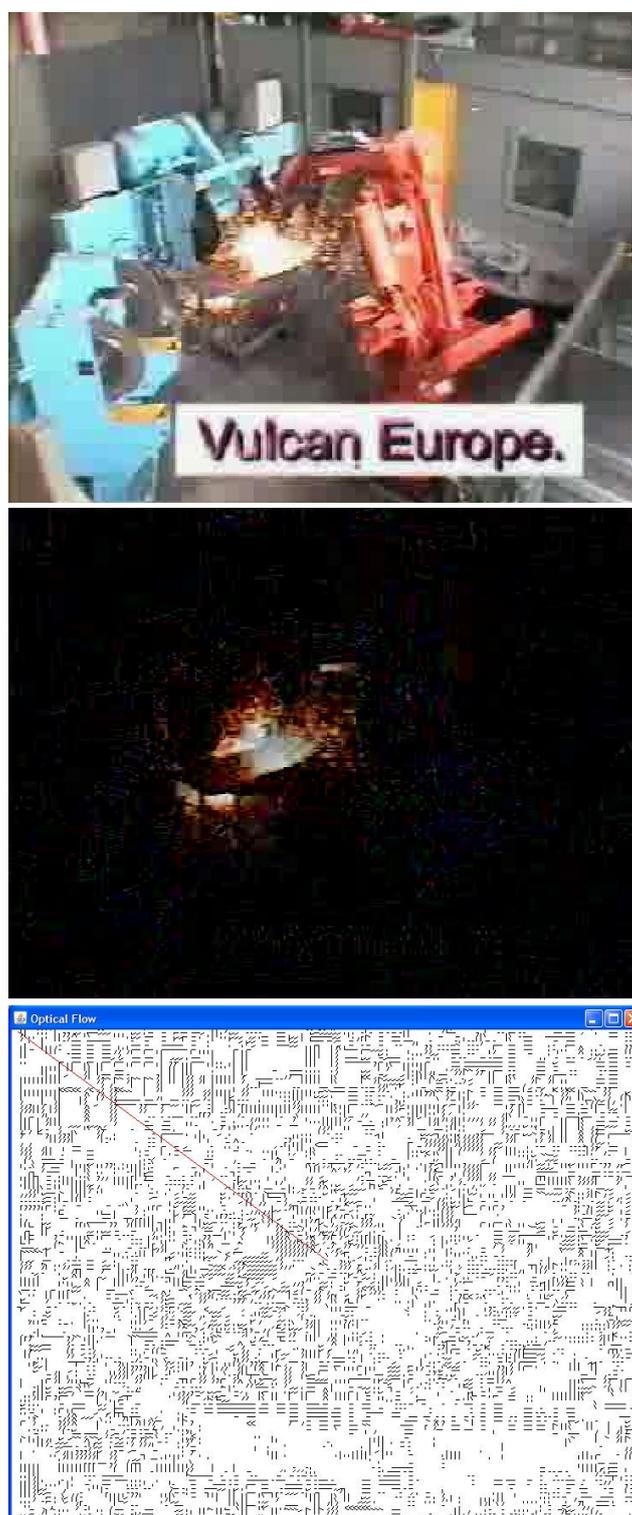


Figura 5.14: Ruido en el gráfico del flujo óptico de una imagen de baja resolución y mala calidad.

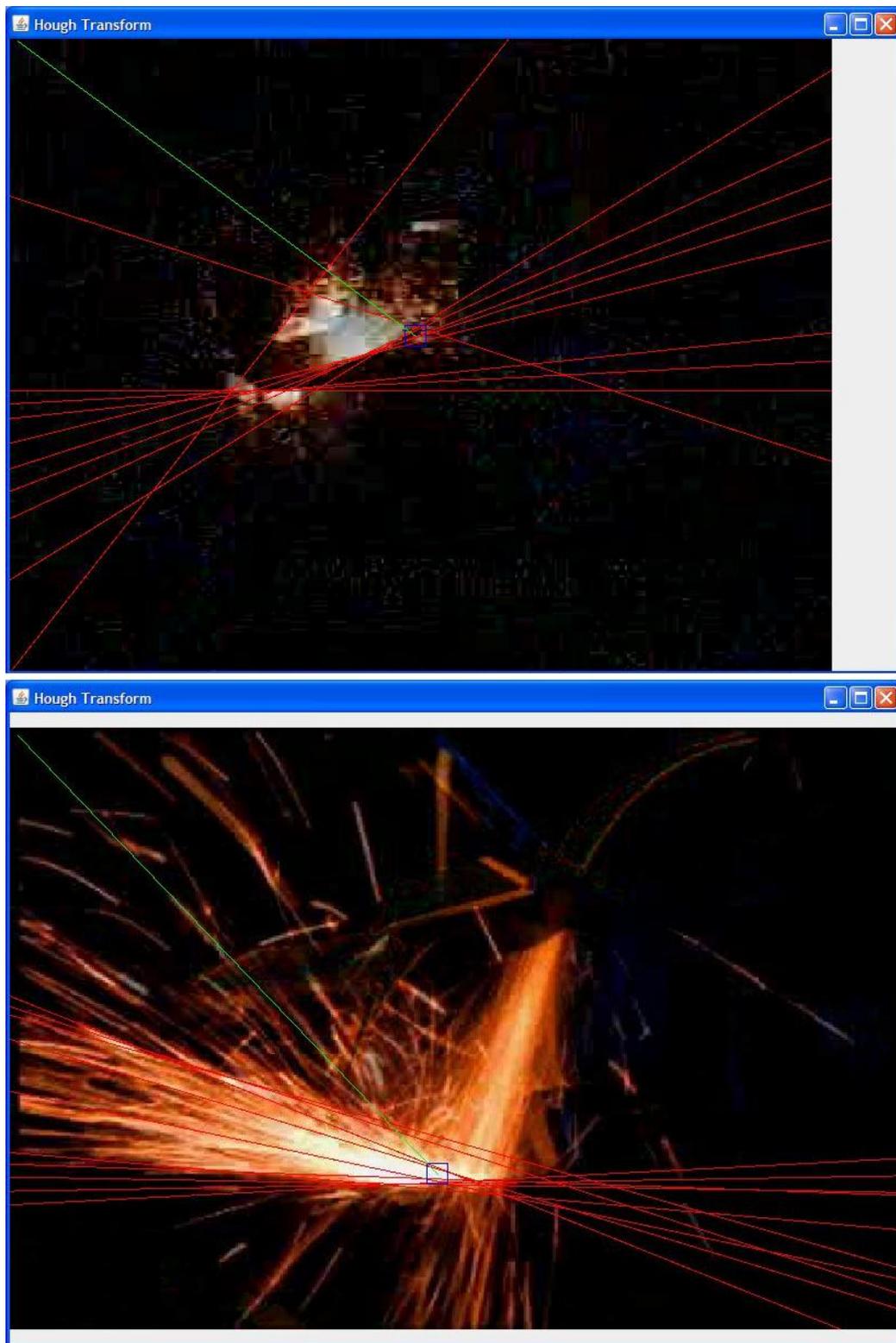


Figura 5.15: Resultado de la integración del flujo óptico y la transformada de Hough.

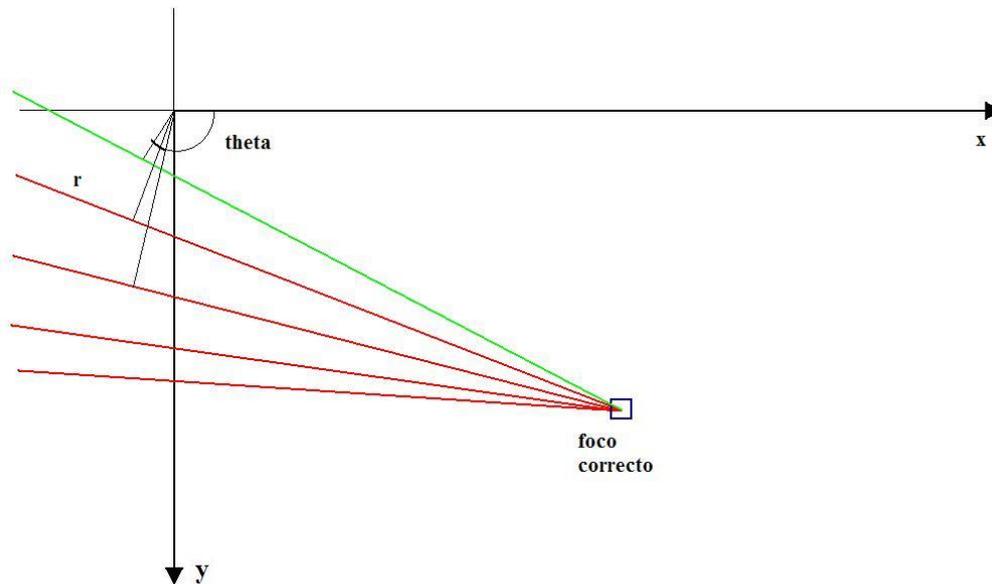


Figura 5.16: Variación de las coordenadas r y θ entre líneas procedentes del mismo punto.

Por ejemplo, en la figura 5.15, se detecta correctamente el foco en ambas ocasiones, pero en ambas se da el aviso debido a que el parámetro θ se aleja de las trayectorias detectadas. En esto influye también el número de líneas detectado, ya que si se tiene mayor número de líneas, es más probable que la línea verde esté entre ellas.

5.6.3. Consideraciones de diseño

Para diseñar esta comprobación, nos hemos basado en la observación de la coordenada θ . Esto se debe a que es la que varía menos entre las trayectorias que parten de un mismo foco. Esto se ilustra en la figura 5.16.

En la figura 5.17 se ilustra la variación de θ entre líneas procedentes de dos puntos diferentes.

Además se ha optado por unir el punto de máxima velocidad con el foco, y no por prolongar el vector de la velocidad de dicho punto hacia el foco. Esto se ha debido a que en las imágenes analizadas se obtenía que el punto de máxima velocidad se correspondía con una chispa en su caída hacia el

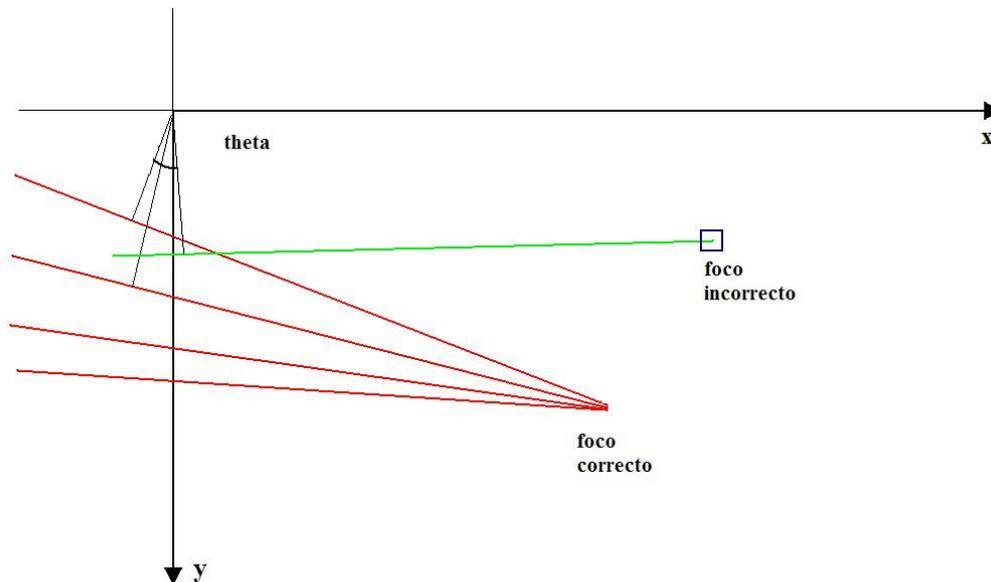


Figura 5.17: Variación de la coordenada theta entre líneas procedentes de puntos diferentes.

suelo. Por eso la prolongación de su velocidad no nos llevaba al foco, y se desechó esa opción.

5.7. Análisis de vídeo

SparksAnalyzer permite configurar las características del trabajo con vídeo, además de la precisión del algoritmo del flujo óptico. Con esto se consigue regular la velocidad del análisis y su precisión.

Como se ha señalado anteriormente, el algoritmo del flujo óptico tiene un gran coste computacional, lo que hace que el funcionamiento de SparksAnalyzer se ralentice. Por tanto si se desea conseguir mayor velocidad se puede reducir la resolución de las imágenes con las que trabaja el algoritmo. También se puede configurar la tasa de procesamiento de las frames que llegan del flujo de vídeo. Por ejemplo si el vídeo tiene una tasa de 30 frames por segundo, se puede optar por procesar una de cada 10. Esto depende del objetivo del análisis. Para un análisis de entrenamiento en el que se busque obtener la mayor cantidad de datos posibles sin importar el tiempo, se

pueden procesar todas las frames al máximo detalle. En cambio si se busca ajustar los parámetros para lograr funcionar en el tiempo de duración del vídeo, habrá que sacrificar detalles y saltar algunas frames.

En cuanto al funcionamiento de los algoritmos, el primer paso que se realiza en el procesamiento de las frames es la sustracción del fondo. Como se mencionó en el apartado 5.1, conviene que el fondo sea estático. Por tanto el funcionamiento de la aplicación será mejor cuanto menos varíe el entorno con respecto al fondo. Por ejemplo, si la primera frame del vídeo, que es la que se toma como fondo, incluye al brazo robótico en una posición que posteriormente va a variar, se obtendrá un resultado peor que si no aparece el brazo. También el resultado será mejor cuanto menos se mueva el brazo.

Por todo lo mencionado anteriormente, la aplicación permite gran flexibilidad de funcionamiento gracias a los numerosos parámetros de configuración, pero la calidad de los resultados depende de la calidad de las imágenes obtenidas y de la localización de los elementos del entorno de trabajo.