

Detección de campo, balón y jugadores de fútbol

Iván Folgueira - ivan.folgueira@udc.es

El problema consiste en detectar el campo de juego y separar sus dos tonalidades de verde, identificar a las personas en el terreno de juego y el balón. Para ello el primer paso será aislar el campo del resto de la imagen en la que también se incluyen las gradas, carteles publicitarios, el marcador de juego, etc. Para lograr esto partimos de un conocimiento básico del dominio tras analizar las distintas imágenes, el campo es de un color verde más o menos uniforme dependiendo de la iluminación y ocupa una parte importante de la imagen, normalmente más del 50%. Con esto en mente se han considerado y probado varias técnicas:

-Máscaras de color: partiendo de que el verde (campo) ocupa una parte importante de la imagen y por lo tanto los píxeles con los valores que forman este color tendrán mayor presencia que otros, hacemos la mediana de los colores de la imagen para cada canal por separado (RGB). De esta forma en una imagen más clara por una mejor iluminación los tonos verdes del campo serán también más claros, lo mismo ocurre con una más oscura. A continuación, creamos 2 umbrales uno con un valor un poco más bajo que la mediana y otro un poco más alto, y nos quedamos con los píxeles de la imagen que estén en estos rangos. De esta forma conseguimos en casi todos los casos la totalidad del terreno de juego, también conseguimos “ruido” de las zonas de las gradas y los otros elementos de la imagen.

-Método Otsu y Otsu múltiple: un intento sin éxito ha sido el método de Otsu, en este método la idea era la de sacar el elemento campo ya que supone una parte más grande de la imagen y se hace de manera automática sobre el histograma. Pero como las imágenes presentan distintos tipos de iluminación y zoom, y además sus histogramas no presentan “picos” evidentes, sino que presentan una gran cantidad de niveles de gris en una región y, en ciertas imágenes, muy uniforme el resto o con otros picos de gris. Debido a esta disparidad en algunos casos los resultados, pese a no ser perfectos eran aceptables, pero en otros directamente no es capaz de identificar el campo correctamente. Para tratar de solucionar esto se intentó hacer lo mismo, pero usando una implementación de Otsu múltiple ya que aquí tenemos varios umbrales distintos. El resultado sin embargo no mejoro mucho ya que, si presentaba mejor separación, pero solo en las imágenes bien clasificadas por el Otsu simple.

Por este motivo para el recorte del campo decidimos proceder con la máscara de color. La salida de esta máscara, al quedarnos con los tonos verdes de la imagen, también presenta ruido en forma de pequeños píxeles y artefactos en la zona de las gradas, para limpiar un poco la imagen antes de proceder aplicamos un operador morfológico de apertura. Esto nos quita muchos elementos, pero aún quedan, también se presentan bordes irregulares en la imagen, para suavizar los bordes que se generan tanto en la máscara como con el operador aplicamos un filtro de mediana para quitar el ruido. La decisión de proceder con un filtro de mediana es por los píxeles de lo que podríamos clasificar como ruido sal-pimienta, en la imagen adjunta “campo prefiltro” se puede ver como en el recorte generado por la máscara de verde quedan bordes irregulares, ruido pequeño en la parte superior y líneas con bordes de sierra. Tras aplicar el filtro estas líneas se fusionan y buena parte del ruido desaparece. En la imagen resultado aparece también el caso aplicando un filtro de gauss (“median y gauss 9x9”), pero en este se puede observar, como es propio de este filtro que solo difumina, pero no corrige realmente estos apartados de la imagen, por ello procedemos con el de medianas.

A partir de la imagen filtrada para sacar el recorte de la imagen de salida donde está el campo y los jugadores sacamos los contornos. Con los contornos obtenidos nos quedamos el mayor que se corresponde con el campo y aplicamos una máscara con el contorno relleno a la imagen de entrada obteniendo de esta forma el recorte completo sin huecos internos, estos huecos serían los producidos por las líneas del campo, los jugadores, el balón... Una vez obtenido el recorte del campo detectaremos en él los jugadores, se han explorado varias opciones:

-La primera y más exitosa ha sido aprovechar los contornos que ya habíamos buscado y quedarnos con los interiores, es decir los que están dentro del contorno grande, el campo, para cada uno de los contornos establecemos unos filtros de dimensiones para descartar aquellos que son muy pequeños y que representan pequeños fallos. Estos fallos podrían ser por ejemplos líneas blancas del césped o pequeñas zonas de ruido en la imagen. Establecemos que estos filtros sean para personas, como las personas son las altas que anchas este será un criterio, pero tampoco muy altas. El criterio sería $1.2 < \text{alto/ancha} < 3.6$.

-Otra opción que se ha intentado, pero presenta más fallos (imágenes adjuntas "HOG personas X") es el uso de HOG (Histogram of Oriented Gradients) para detección de personas. Se ha intentado este método ya que es bueno para buscar personas en imágenes, pero en nuestro caso no presenta buenos resultados detectando a lo sumo 1 o dos personas en una imagen con 7-8.

Teniendo esto en cuenta nos quedamos con la primera opción ya que presenta mejores resultados en nuestro caso concreto y no requiere tanta carga computacional.

Por último, para la detección de las "líneas" de color del césped también se han intentado un par de opciones:

-La primera de ellas fue usar K-Means a partir de la imagen suavizada anteriormente con 4 clusters ya que estas imágenes tienen 4 colores principales, (negro, una tonalidad de verde, la otra tonalidad y uno más para los pequeños pixeles de ruido que no entran en ninguna de las anteriores) esto mostró resultados dispares en función de la imagen de entrada y como esta estuviese iluminada, (imágenes "kmeans X"). Los resultados de kmeans se pasarían por un detector de bordes, Canny. Pero ya el primer paso no muestra buenos resultados en general. La idea de intentar k-means parte de que los colores de las líneas del césped están relativamente bien definidos, pero con muchos niveles de color diferentes por en medio y k-means los agruparía para el detector.

-El segundo intento, y el que finalmente se usa, es combinar un par de métodos, primero a partir de la imagen suavizada de antes la convertimos a escala de grises y realizamos los bordes mediante unsharp-mask, ya que tras probar sin el realce apenas se detectaban los bordes necesarios. Sobre esta imagen aplicamos detector de bordes de Canny que al haber hecho los pasos anteriores saca unos bordes decentes para la mayoría de las imágenes, la elección de Canny es porque necesitamos bordes finos y salida binaria, y Canny proporciona ambas. Sobre esta imagen binaria con los bordes de Canny sacamos las líneas rectas con la transformada de Hough, que pese a fallar en ciertos casos los resultados son bastante aceptables en otros. Optamos por la transformada de Hough porque permite obtener formas y figuras geométricas como las líneas que es lo que nos interesa. Tras extraer muchas líneas, ya que ponemos unos criterios poco estrictos para poder filtrar manualmente, filtramos manualmente los resultados quedándonos solo con las que son líneas diagonales o verticales, dados los requisitos del problema. Por último, representamos todo esto en una imagen final y la mostramos.

Problemas sin resolver y mejoras:

- identificación del balón: con la implementación actual no es posible, pero se podría usar Hough para círculos, por ejemplo.

- mejorar la detección de líneas del césped: jugando con múltiples parámetros de Hough, añadiendo filtros a los resultados, etc.

- mejorar el recorte del campo: en algunas imágenes no es bueno se podría mejorar con mallas topológicas activas o seguimiento de contornos, por ejemplo. Lo mismo para los jugadores.