Reconocimiento gestual con Kinect para detectar comportamientos inseguros en conductores

Gesture recognition using Kinect to detect unsafe behaviors performed by drivers

Diana Esquivel-González[1]

Esquivel-González, D. Reconocimiento gestual con Kinect para detectar comportamientos inseguros en conductores. *Tecnología en Marcha*. Vol. 33, especial Movilidad estudiantil. Pág 166-175.

https://doi.org/10.18845/tm.v33i7.5491

1 Ingeniera Mecatrónica. Desarrollado en: Laboratório de Robôtica Móvel, ICMC, Universidade de São Paulo, Brasil. Área Académica Mecatrónica, Tecnológico de Costa Rica. Correo electrónico: dianaesquivel100@hotmail.com

Palabras clave

Kinect; “Point Cloud”; Sensor de profundidad; Substracción de fondo; Reconocimiento gestual; Procesamiento de imágenes.

Resumen

Los comportamientos inseguros por parte del conductor de un vehículo, como el uso de un dispositivo celular durante la conducción, son unas de las principales causas de accidentes de tránsito alrededor del mundo. Esta práctica es difícil de regular y controlar, mas con el uso de un sistema automatizado de detección de gestos que indique un comportamiento peligroso, es posible. El uso de cámaras con sensores de profundidad en 3D permite analizar objetos y personas en tiempo real con una alta precisión, identificando rasgos y figuras. Estos sensores utilizan una tecnología denominada “point cloud”, la cual emite un arreglo de luz láser y mide el tiempo de regreso hacia el sensor, determinando la profundidad de cada uno de los puntos. Esta tecnología permitió realizar una aplicación de reconocimiento de gestos que es capaz de detectar de manera automática la acción que indica que el conductor está utilizando un dispositivo celular durante el tiempo de conducción. Utilizando algoritmos de substracción de fondo, análisis de rasgos y análisis de histogramas se consiguió realizar inicialmente una simulación funcional en el programa V-Rep que es capaz de detectar el gesto que indica este comportamiento. Posteriormente, se implementó un prototipo utilizando imágenes reales tomadas con un Kinect que fueron procesadas en el programa Octave con los mismos algoritmos utilizados en la simulación, para así probar la efectividad de la aplicación desarrollada.

Keywords

Kinect; “Point Cloud”; Depth sensor; Background Subtraction; Gesture Recognition; Image Processing.

Abstract

Vehicle drivers’ unsafe behaviors, like using a cellular phone while driving, is one of the leading causes of traffic accidents worldwide. This unsafe practice is hard to regulate and control. However, it is possible with the use of an automated gesture detection system that indicates a dangerous behavior. The use of cameras with 3D depth sensors can aid in analyzing objects and people in real time with high precision, identifying features and figures. These sensors use a technology referred to as “point cloud”, which emits an array of laser lights and measures the time of return (fly time) back to the sensor, determining the depth of each one of these points. This technology permitted the development of a gesture recognition application that is capable of automatically detecting the action that indicates that a driver is using a cellular phone while driving. An initial functional simulation was developed using background subtraction, feature analysis, and histogram analysis algorithms using the program V-Rep, which can detect the gesture that would indicate this behavior. Furthermore, a prototype was implemented using real images taken with the Kinect that were processed using Octave using the same algorithms used for the simulation, thus supporting the effectiveness of the application.

Introducción

La conducción inapropiada e insegura de automóviles genera millones de accidentes al año en todo el mundo. Según la Organización Mundial de la Salud, alrededor de 1.3 millones de personas mueren cada año en accidentes de tránsito (Organización Mundial de la Salud, 2009).

Gran parte de los accidentes que suceden, ya sea en la industria o en las carreteras, se deben a error humano. Este factor puede eliminarse con la automatización, ya que las máquinas y las computadoras no cometen errores y no sufren de características humanas como el cansancio y la distracción.

Un problema común que ha sido regulado por la ley pero que sigue siendo un problema es la de utilizar dispositivos móviles, como el celular, mientras se está conduciendo. Este acto es peligroso debido a que desvía la atención del conductor y evita que esté alerta a las situaciones de riesgo que suceden a su alrededor. A pesar de ser un acto penalizado por ley, es difícil poder detectar cuando se da una infracción y aún más difícil poder probar que efectivamente se estaba haciendo uso del dispositivo en el momento en que la persona iba conduciendo.

A pesar de que se han desarrollado aplicaciones automatizadas para detectar acciones inseguras y para prevenir accidentes de tránsito, aún no se cuenta con una solución comercial y confiable para detección del acto de hablar por celular mientras se conduce. Muchos de los proyectos que se han trabajado en este tema lo ven como un problema secundario al que abordan de manera superficial, por lo que no se ha logrado encontrar una solución viable al problema que se plantea.

En el presente artículo se expone la elaboración de una solución para dicha aplicación, que logra detectar de manera automatizada el uso del celular. Utilizando algoritmos para sustracción de fondo, análisis de rasgos de la figura humana y análisis de histogramas, se desarrolló una simulación en V-Rep y posteriormente una implementación de manera de prototipo utilizando las imágenes obtenidas con el sensor Kinect y analizadas en Octave.

Consideraciones para reconocimiento de gestos

Hay diversos pasos que se deben seguir para realizar un adecuado reconocimiento de gestos. El primero es tener un modelo de gestos, para el cual se debe estudiar las características del gesto que se quiere reconocer. Una vez que se tiene el modelo, se debe realizar un análisis para definir los parámetros que se tomarán en cuenta. Cuando se tienen los parámetros bien definidos, se procede a la etapa de reconocimiento, en donde se reconocen los distintos gestos que fueron predefinidos. Finalmente, se utilizan los gestos reconocidos para alguna aplicación o sistema. Para este último paso se le da un uso concreto al reconocimiento de gestos, con el cual se llevará a cabo una acción que está ligada a la información analizada. (Premaratne, 2014).

La complejidad en reconocimiento gestual se debe a la amplia variedad de gestos que pueden existir, y al hecho de que un mismo gesto no producirá la misma serie de imágenes en dos ocasiones distintas. Es por ello que se debe trabajar con modelos de gestos que puedan definir claramente lo que un gesto específico significa en términos de posiciones y movimientos de las manos y de expresiones faciales. Para simplificar la tarea de discernir entre una infinidad de gestos, se trabajó de manera simplificada haciendo detección de la mano y la cabeza, discerniendo entre ellas para detectar los movimientos que se producen.

Elaboración del programa

Se realizó inicialmente una simulación en V-Rep que hace uso de los sensores que se utilizarían para la aplicación real, permitiendo la visualización del comportamiento y la recolección de información del Kinect. Para cada elemento utilizado en el ambiente de simulación (sensores y actuadores) se puede modificar el “script” para así determinar los parámetros de funcionamiento y las condiciones de uso. También se puede simular una persona y regular su movimiento por medio del código que lo define. Otros elementos en la escena (mesa para posicionar el

Kinect, silla para simular el asiento del vehículo, elementos de fondo) se utilizaron para hacer la simulación más cercana a una situación real de uso, y para probar la eficacia del algoritmo de sustracción de fondo.

Se realizaron varias funciones utilizando el sensor Kinect, incluyendo la toma de imagen de fondo, la creación de la máscara que identifica a la figura humana (substracción de fondo), la identificación de la zona de interés (zona de la cabeza) y la identificación del gesto de hablar por celular. La sustracción del fondo, como su nombre lo indica, se encarga de eliminar ciertos elementos para obtener solamente los puntos de interés, restando la imagen actual de una imagen tomada anteriormente que representa el fondo. El resultado de la aplicación de este algoritmo da como resultado una máscara binaria que indica los objetos que se quieren analizar, en este caso la figura de la persona. El archivo resultante que se crea al presionar el botón *Fondo* en el programa es un documento de texto con datos numéricos con 4 posiciones decimales que van del 0 al 1 (0 siendo el elemento más cercano a la cámara y 1 el elemento más lejano). Este archivo tendrá 64 columnas y 48 filas, correspondiendo a la cantidad de pixeles que se tienen por la resolución que se le estableció al Kinect. Es importante actualizar periódicamente la imagen del fondo, para así incluir otros elementos que pudieron entrar en el marco de visión y que realmente no son de interés. (Vacavant, Chateu, Wilhelm, & Lequièvre, 2013)

Para evitar detecciones falsas debido a movimientos leves de la cámara o a cambios en la iluminación, se utilizó un valor umbral al hacer la sustracción de fondo. Para la simulación este valor no es necesario ya que no hay dichas variaciones pero para el prototipo debió determinarse el valor umbral de manera experimental. Además se realizó pre-procesamiento de la imagen, que incluye conversión a escala de grises, suavizado de la imagen y filtros, para eliminar ruido que se pueda generar y facilitar la aplicación de los algoritmos. La substracción de fondo se realiza de manera continua en un ciclo infinito mientras se esté corriendo el programa, con el fin de poder trabajar con la posición actual de la persona y poder hacer una detección apropiada.

La zona de interés con la que se trabajó es el área que va desde el cuello de la persona hasta la punta de la cabeza, ya que al identificar un objeto que se acerca a esta zona durante el tiempo en que la persona está conduciendo, indicaría la presencia de un celular. Antes de poder identificar que existe un objeto en cercanía a la cabeza, se debe determinar el punto en donde comienza y termina ésta, ya que varía con cada persona y con la posición del asiento que establezca la persona al conducir. La identificación de la zona de interés deberá hacerse una única vez y asegurando que la persona se encuentre en posición neutral (con los brazos abajo) para evitar mediciones erróneas. Para tomar estos datos, se estableció de nuevo un botón que almacena la información actual del sensor y a partir de ella realiza las mediciones para encontrar los dos puntos de interés.

Para detectar la forma de la cabeza, se calculó el punto máximo de ésta utilizando la coordenada Y máxima que además pertenezca a la máscara de valores de puntos de interés (punto máximo de la figura de la persona), utilizando la siguiente fórmula:

$$Coord\ Y = tan\left(Ángulo\ Y\right) * Coord\ Z$$

En donde *Coord Z* es el valor de profundidad medido con el Kinect y ÁnguloY es un porcentaje del ángulo de medición del Kinect (en radianes) que se calcula con la siguiente fórmula que utiliza la resolución establecida del Kinect (en ambos ejes):

$$\text{ÁnguloY} = \frac{\text{j} - \frac{\text{ResY}}{2} + 0.5}{\frac{\text{ResY}}{2}} * \text{ÁnguloMedioY}$$

El siguiente paso fue encontrar el punto en donde inicia el cuello. Para ello hay dos métodos que se utilizaron. El primero de ellos consiste en realizar un histograma con la cantidad de pixeles que pertenecen a cada fila horizontal de la máscara. Se sabe que una persona en posición neutral tendrá un ancho casi uniforme a lo largo de su cuerpo hasta llegar al cuello, en donde se verá reducido significativamente el ancho medido (cantidad de pixeles). Para poder leer la cantidad de pixeles por fila, se recorrió cada fila y se incrementó un contador cada vez que se reconoce un pixel que no pertenece al fondo. Al finalizar el recorrido de todas las filas, se obtuvo un vector con todos los valores correspondientes al conteo de cada fila. Comparando la proporción entre el ancho máximo del cuerpo y el ancho de la fila medida se consiguió identificar la posición del cuello.

Otro método que se puede utilizar para identificar el cuello es comparar los valores de profundidad, ya que hay un cambio significativo entre la profundidad medida en la cara y la que se puede medir en el cuello. Esto permite identificar la forma de la cabeza de la persona. Ya que el punto máximo de la cabeza tiene una profundidad mayor que cualquier punto de la cara pero menor que cualquier punto del cuello, se utilizó este valor para detectar el punto de inicio del cuello, que tendría una relación $\frac{Profundidad\ medida}{Profundidad\ del\ punto\ máximo\ de\ la\ cabeza} > 1$. Con una resolución mayor al utilizar un sensor Kinect en una aplicación real, esta ecuación es aún más exacta.

Para detectar la presencia del dispositivo móvil, se analizó el área alrededor de la cabeza para detectar objetos cercanos a ella. Para esta aplicación, cualquier objeto de tamaño significativo que se acerque a la oreja por un tiempo prolongado (5 segundos) es interpretado como un dispositivo móvil. Para la detección, se utilizó el mismo método de análisis de histogramas para detectar un aumento significativo en la cantidad de pixeles medidos por el sensor en las zonas laterales.

Una vez realizada la simulación, se utilizó un sensor Kinect real para obtener datos que permitieran determinar si la aplicación desarrollada realmente es efectiva. Se utilizó un ambiente similar al de la simulación, en donde se tienen algunos elementos que forman parte del fondo y una persona en posición sentada. Para ello, se utilizó el software de Microsoft SDK con un sensor Kinect II (Xbox One), y se tomó una serie de 80 imágenes (en formato .bmp) que incluyen el fondo sin personas en la escena e imágenes de la persona sentada con los brazos en posición neutral y utilizando el celular con cada una de las manos.

Resultados

Para la simulación realizada se obtuvieron resultados favorables utilizando ambos métodos.

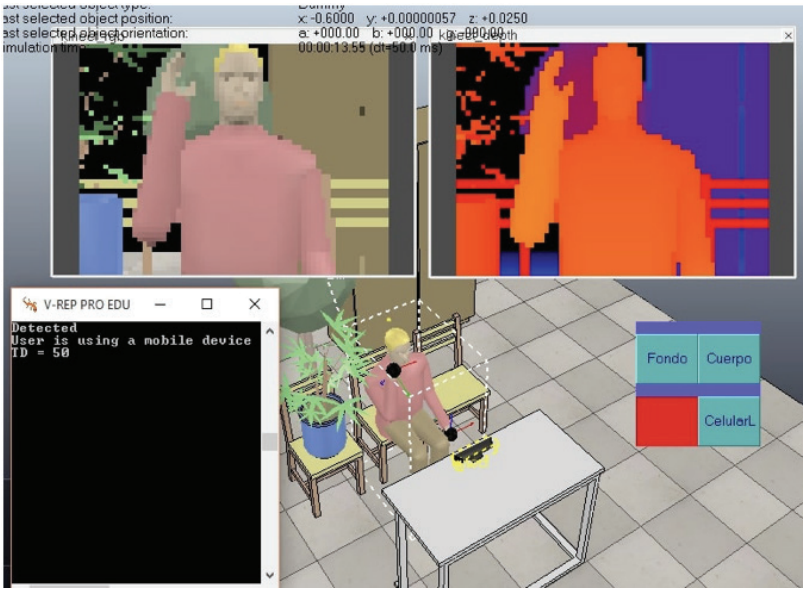

Figura 1: Detección de gesto de uso de dispositivo móvil (Creado por el autor).

En la figura 3 se puede observar cómo, al tener a la persona hablando por celular, el programa detecta la mano en cercanía a la cabeza e imprime un mensaje de detección en pantalla. Este mensaje se imprime aun cuando la detección se ha dado por un tiempo muy corto, pero no activará una alarma hasta que se alcancen los 5 segundos. Al detectarse la acción de hablar por celular por más de 5 segundos, se imprime el segundo mensaje en pantalla “El usuario está usando un dispositivo móvil” y se activa una alarma sonora para alertar al conductor. Se puede observar que el tiempo de detección en número de ciclos es de 50, lo cual equivale a 5 segundos.

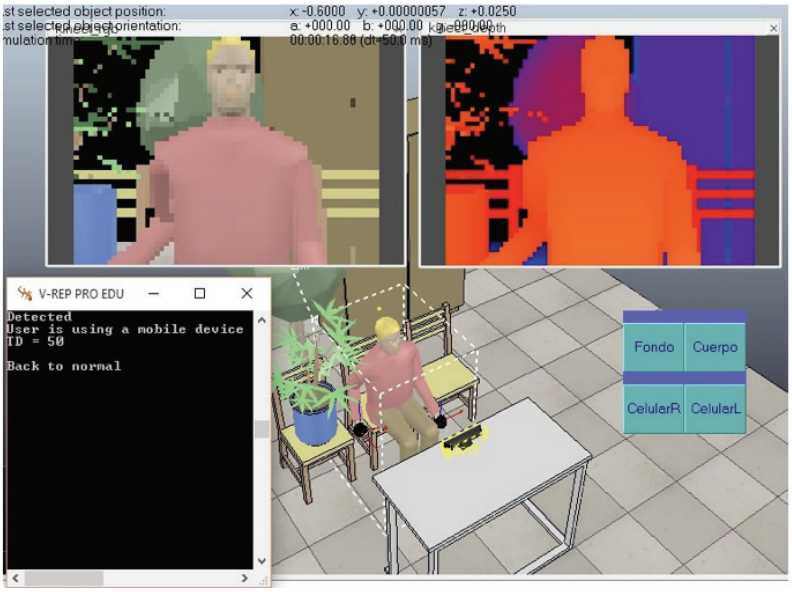

Figura 2: Detección de final de gesto (Creado por el autor).

Finalmente, al regresar a la posición neutral, el programa reconoce que ya no hay ningún objeto cercano a la cabeza que pueda representar un acto inseguro. Cuando el programa deja de detectar el gesto, se imprime un mensaje en pantalla “Vuelta a la normalidad”, indicando que el usuario ya no corre peligro y que el contador ha vuelto a cero.

Para los datos del Kinect se realizó primero un pre-procesamiento de suavizado de las imágenes. Se obtuvo un promedio de imágenes que representan el fondo, y un promedio de imágenes de la persona en posición neutral, para así realizar la sustracción de fondo y facilitar el procesamiento en Octave.

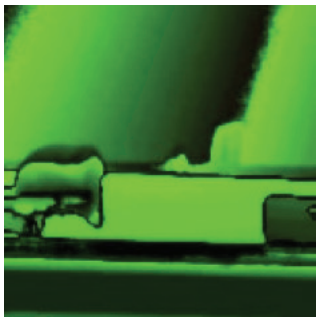

Figura 3: Imagen de fondo con suavizado (Creado por el autor).

Figura 4: Persona en posición neutral con suavizado (Creado por el autor).

Utilizando los algoritmos utilizados en la simulación y con un valor umbral determinado experimentalmente, se obtuvieron los siguientes resultados:

Figura 5: Detección de objetos de interés después de sustracción de fondo (Creado por el autor).

Aunque hay algunos espacios en negro, se puede detectar claramente el contorno de la figura humana, y con esta imagen se realizó el análisis de los contornos para encontrar los puntos de interés (punto máximo de la cabeza, extremos laterales, posición del cuello). Se utilizaron una variedad de imágenes en diferentes posiciones: neutral, con las manos al frente, sosteniendo el celular con la mano izquierda y con la derecha, para probar los algoritmos y realizar la detección.

Figura 6: Imágenes para análisis en diferentes posiciones después del pre-procesamiento y sustracción de fondo.

Para la posición neutral de la persona, el programa determinó que no hay un dispositivo móvil presente y por tanto, las condiciones de conducción son seguras. Este mismo resultado se obtiene para la imagen con las manos al frente.

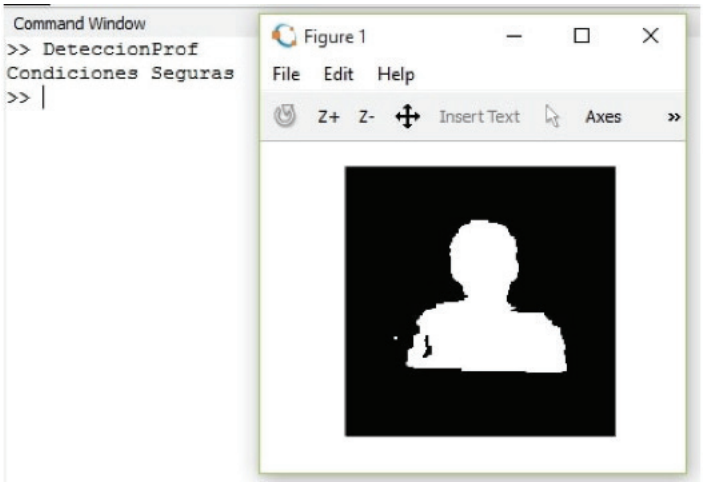

Figura 7: Análisis de imagen en posición neutral (Creado por el autor).

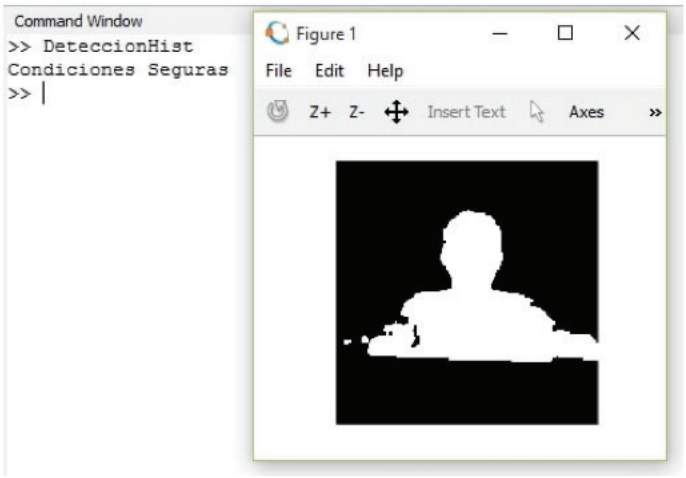

Figura 8: Análisis de imagen con manos al frente (Creado por el autor).

Finalmente, al correr el programa con las imágenes en donde se hace uso del celular, se detectó la presencia del mismo y se imprimió un mensaje de alerta para el usuario, indicando que la conducción no es segura.

Figura 9: Análisis de imagen utilizando dispositivo móvil (Creado por el autor).

Conclusiones

Dados los resultados obtenidos se puede concluir que se consiguió realizar exitosamente el diseño de una aplicación de reconocimiento gestual que permitiera reconocer el gesto de hablar por celular de manera automatizada. Asimismo, se realizó una simulación funcional utilizando dos métodos distintos con el programa V-REP con los cuales se obtuvieron resultados deseables. Posteriormente, se utilizó exitosamente los datos de un sensor real para respaldar los resultados obtenidos en la simulación con un programa funcional utilizando dos métodos distintos con el programa Octave. Al igual que en la simulación, se obtuvieron resultados

deseables de la parte de verificación de datos, realizando detección de gesto correctamente con ambos métodos.

Aunque este desarrollo funcionó adecuadamente, hay espacio para ampliar y mejorar los algoritmos de detección que permitan una solución más robusta. Se debe considerar las variaciones en iluminación y movimiento, las variaciones anatómicas de las personas y la presencia de obstáculos (cabello y accesorios) para poder obtener datos más confiables y realizar detecciones más exactas. Asimismo, se debe ampliar la aplicación para detectar más concretamente la presencia de un dispositivo móvil (contrario a detección únicamente de un objeto en cercanía a la oreja). La detección de gestos utilizando visión computacional permite la investigación y desarrollo en una variedad de aplicaciones que permitan mejorar la seguridad y a experiencia de usuario.

Referencias

[1] T. Bouwmans, F. Porikli, B. Höferlin, & A. Vacavant. (2014). *Background Modeling and Foreground Detection for Video Surveillance.* Chapman and Hall/CRC.

[2] S.-C. S. Cheung, & C. Kamath. (2007). *Robust techniques for background subtraction in urban traffic video.* Technical Paper, Center for Applied Scientific Computing, Lawrence Livermore National Laboratory, California. Recuperado el 27 de setiembre de 2015, de https://computation.llnl.gov/casc/sapphire/pubs/UCRL-CONF-200706.pdf

[3] Dramanan. (2013). Background Subtraction. Recuperado el 27 de setiembre de 2015, de http://www.ics.uci.edu/~dramanan/teaching/cs117_spring13/lec/bg.pdf

[4] E. Murphy-Chutorian, A. Doshi, & M. M. Trivedi. (2007). Head Pose Estimation for Driver Assistance Systems: A Robust Algorithm and Experimental Evaluation. *Intelligent Transportation Systems Conference, 2007* (pp. 709-714). IEEE. Recuperado el 22 de setiembre de 2015, de http://ieeexplore.ieee.org/xpl/login.jsp?tp=&arnumber=4357803&url=http%3A%2F%2Fieeexplore.ieee.org%2Fstamp%2Fstamp.jsp%3Ftp%3D%26arnumber%3D4357803

[5] Open CV. (n.d.). *Open Source Computer Vision.* Recuperado el 26 de setiembre de 2015, de How to Use Background Subtraction Methods: http://docs.opencv.org/master/d1/dc5/tutorial_background_subtraction.html#gsc.tab=0

[6] Organización Mundial de la Salud. (2009). *Informe sobre la situación mundial de la seguridad vial 2009.* Recuperado el 29 de agosto de 2015, de http://www.who.int/violence_injury_prevention/road_safety_status/report/web_version_es.pdf?ua=1

[7] Premaratne, P. (2014). Historical Development of Hand Gesture Recognition. En *Human Computer Interaction Using Hand Gestures* (págs. 5-29). Singapore: Springer. doi:10.1007/978-981-4585-69-9_2

[8] A. Vacavant, T. Chateu, A. Wilhelm, & L. Lequièvre. (2013). A benchmark dataset for outdoor foreground/background extraction. *Computer Vision-ACCV 2012 Workshops*, (págs. 291-300). Springer. doi:10.1007/978-3-642-37410-4_25

[9] Zhang, C., Yang, X., & Tian, Y. (2013). *Histogram of 3D Facets: A Characteristic Descriptor for Hang Gesture Recognition.* New York: IEEE. Recuperado el 16 de octubre de 2015, de http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6553754