

Envío: 16-06-2012

Aceptación: 18-06-2012

Publicación: 19-06-2012

SISTEMA DE RECONOCIMIENTO FACIAL Y REALIDAD AUMENTADA PARA DISPOSITIVOS MÓVILES

**FACIAL RECOGNITION AND AUGMENTED REALITY SYSTEM
FOR MOBILE DEVICES**

Juan Vicente Martínez Pérez¹

Jordi Linares Pellicer²

1. Ingeniero Técnico en Informática de Gestión. Instituto Tecnológico de Informática. Universitat Politècnica de València.
2. Doctor en Informática. Instituto Tecnológico de Informática. Universitat Politècnica de València.

RESUMEN

El objetivo de este artículo es explicar la problemática de utilizar las funciones de reconocimiento facial en los dispositivos móviles actuales, así como dar una posible solución basándonos en un diseño cliente-servidor.

ABSTRACT

The aim of this paper is to explain the problems of using facial recognition features on today's Mobile devices and provide a possible solution based on a client-server design.

PALABRAS CLAVE

Reconocimiento facial, realidad aumentada, móviles, opencv, cliente-servidor.

KEYWORDS

Facial recognition, augmented reality, mobile, opencv, client-server.

INTRODUCCIÓN

En los últimos años, el desarrollo de nuevo Hardware y Software informático para sistemas de seguridad ha experimentado un gran impulso, tal es el caso de los sistemas de reconocimiento por huella dactilar, voz, iris y facial. Entre estos, destaca por perfilarse como el más prometedor el reconocimiento facial.

La identificación de características faciales ^[1] ha recibido un fuerte impulso gracias al avance en la tecnología de vídeo multimedia propiciándose así un aumento de cámaras en los lugares de trabajo, hogar y dispositivos móviles con un reducido coste. El reconocimiento facial se puede aplicar en el control de accesos a edificios públicos y privados, cajeros automáticos, laboratorios de investigación, como clave secreta de acceso para el uso de ordenadores personales o terminales móviles de última generación así como para servir de tarjeta de visita de una persona.

El proceso de identificación facial se divide básicamente en dos tareas: detección y reconocimiento ^[2]. La primera de ellas, la detección, comprende la localización de una o varias caras dentro de una imagen, ya sea fija o una secuencia de vídeo. La segunda tarea, el reconocimiento, consiste en la comparación de la cara detectada en el paso anterior con otras almacenadas previamente en una base de datos. Estos procesos, detección y reconocimiento, no deberían ser totalmente independientes debido a que según la forma en la que se detecte una cara puede ser prácticamente imposible su reconocimiento con caras de una base de datos detectadas de manera diferente, de ahí que los sistemas de reconocimiento facial estén fuertemente condicionados por la posición y orientación de la cara del sujeto con respecto a la cámara y las condiciones de iluminación en el momento de realizar la detección.

MOTIVACIÓN

Dependiendo del algoritmo de reconocimiento facial que utilicemos, ésta tarea puede suponer una carga muy elevada para el dispositivo que la lleve a cabo, pudiendo consumir prácticamente todos los recursos disponibles del sistema y dejándolo inutilizable durante el tiempo de proceso.

Es por ello necesario buscar una solución a este problema de cara a aquellos dispositivos que no cuentan con dicha capacidad de cálculo y requieran ejecutar este tipo de aplicaciones multimedia.

ANÁLISIS

Tras realizar un análisis exhaustivo de diferentes métodos de reconocimiento y la forma de llevar a cabo dicha tarea en dispositivos con escasa capacidad de cálculo ^[3], se ha pensado que la mejor opción es hacer una división de tareas del procedimiento completo. Así, el sistema se va a dividir en 2 pilares fundamentales:

- Un dispositivo de detección
- Un servidor de reconocimiento

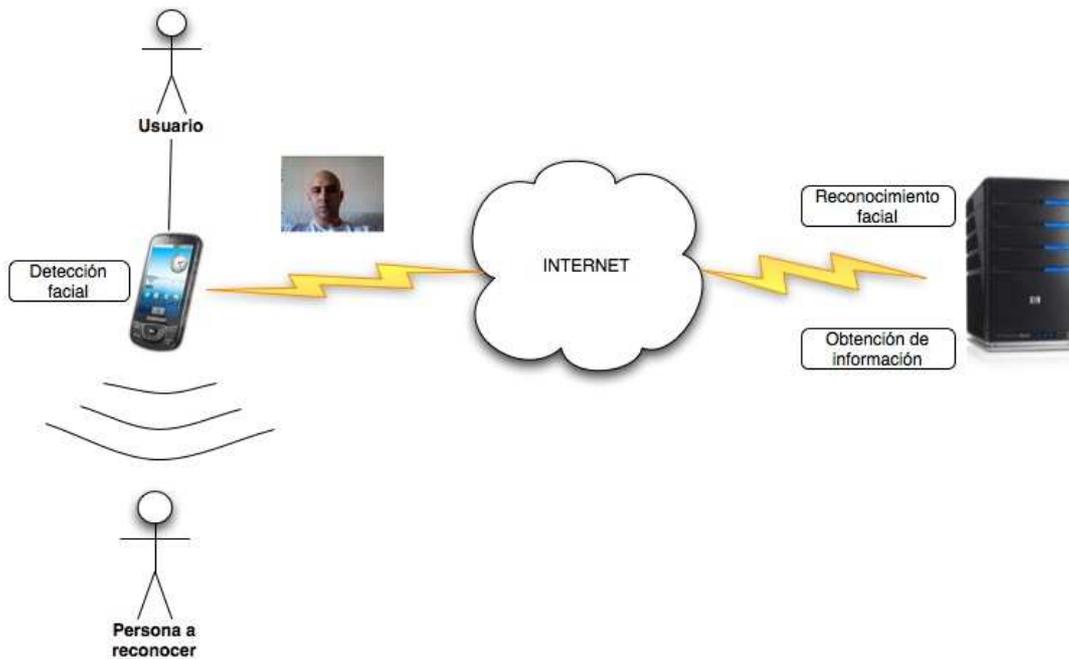


Imagen 1. Diseño del sistema. Fuente: elaboración propia.

Hoy por hoy, todo dispositivo móvil dispone de una conexión de red, ya sea Wifi o cableada, por lo que la implementación de un modelo basado en cliente-servidor puede ser una solución aceptable.

El dispositivo de detección es el encargado de capturar la imagen o vídeo que contiene a la persona cuya cara se quiere reconocer. Dicho dispositivo cuenta con una cámara, integrada o externa. Una vez capturada la imagen, detecta en ella las caras de la persona o personas a las que se esté enfocando, se extraerá dicha región y se enviará a través de la conexión disponible en ese momento al servidor de reconocimiento. Resumiendo:

- Capturar la imagen
- Detectar las caras
- Enviar la región seleccionada al servidor y esperar respuesta

Por otra parte, el servidor de reconocimiento se está ejecutando en modo pasivo a la espera de recibir conexiones provenientes de los dispositivos que ejecutan el software cliente e interactúa con ellos para satisfacer las necesidades oportunas. Las tareas del servidor de reconocimiento son:

- Preprocesar la imagen que ha recibido del cliente
- Realizar el reconocimiento y obtener información extra
- Mejorar las características del clasificador
- Dar respuesta al cliente

Como puede apreciarse, las tareas del cliente son tareas livianas que podrá realizar sin consumir excesivos recursos del sistema.

DETECCIÓN FACIAL

La detección facial es el proceso de encontrar una cara en imágenes o videos. El algoritmo de detección de caras está basado en una función que busca regiones rectangulares dentro de una imagen, regiones que contengan objetos que con una alta probabilidad se parezcan a otros de un conjunto de entrenamiento, devolviendo la región rectangular de la imagen donde se han encontrado. La función escanea varias veces la imagen y con diferentes escalas para encontrar objetos parecidos pero de diferentes tamaños. Por tanto, para detectar caras, únicamente hay que pasarle a la función el conjunto de caras de entrenamiento con las características deseadas para que las caras detectadas sean parecidas a estas.



Imagen 2. Detección facial. Fuente: elaboración propia.

El detector de objetos utilizado en éste artículo fue inicialmente propuesto por Paul Viola y mejorado por Rainer Lienhart ^[4]. En primer lugar, un clasificador en cascada (es decir, una cascada de clasificadores impulsados trabajando con características haar-like ^[5]) es entrenado con unos cientos de imágenes de prueba de un objeto particular (por ejemplo una cara, un coche, etc.), llamadas imágenes positivas que son escaladas al mismo tamaño; e imágenes negativas (imágenes arbitrarias del mismo tamaño).

Después de que el clasificador haya sido entrenado, puede ser aplicado a una región de interés (del mismo tamaño que el usado durante el entrenamiento) de una imagen de entrada. El clasificador devuelve un “1” si la región contiene el objeto, y “0” en otro caso. Para buscar el objeto en la imagen completa se mueve la ventana de búsqueda a través de la imagen y se comprueba cada localización usando el clasificador. El clasificador está diseñado para que pueda ser fácilmente redimensionado para que sea capaz de encontrar los objetos de interés en diferentes tamaños, lo cual es mucho más eficiente que redimensionar la propia imagen. Por tanto, para encontrar el objeto de un tamaño no conocido en la imagen, el procedimiento de escaneado debe ser repetido varias veces con diferentes escalas.

La palabra “cascada” en el nombre del clasificador significa que el clasificador resultante consiste en varios clasificadores más simples (o etapas) que son aplicados uno tras otro a una región de interés hasta que en alguno de los pasos el candidato es rechazado o todos los pasos son satisfactorios. La palabra “impulsados” significa que los clasificadores en todas las etapas de la cascada son complejos y se construyen a partir de los clasificadores básicos usando una de las cuatro técnicas diferentes de impulso (voto ponderado). Los clasificadores básicos son clasificadores en árboles de decisión con al menos 2 hojas.

RECONOCIMIENTO FACIAL

El reconocimiento facial es un área de investigación muy activa especializada en cómo reconocer caras en imágenes o vídeos. El reconocimiento facial es el proceso de hacer corresponder la cara detectada a una de las muchas caras conocidas por el sistema de ficheros.

Existen multitud de algoritmos disponibles para llevar a cabo el reconocimiento facial de entre los que se destacan ^[2]:

- Eigenfaces o método de Análisis de Componentes Principales (PCA)
- Fisherfaces o método de Análisis Lineal Discriminante
- Métodos Kernel
- Métodos de reconocimiento facial 3D
- Método de Gabor Wavelets
- Modelos ocultos de Markov
- Modelos de Apariencia Activa

Para éste desarrollo se ha utilizado el método de las Eigenfaces ^[6] por ser uno de los más simples de implementar.

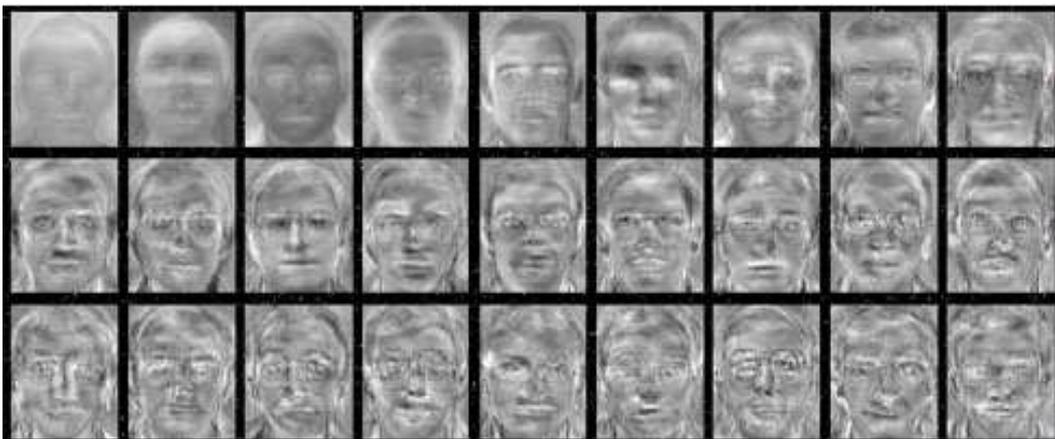
El proceso de reconocimiento facial mediante el uso de la descomposición en valores principales (PCA) y Eigenfaces, sin entrar mucho en detalles, consta de los siguientes pasos:

- Almacenar un conjunto de imágenes de entrenamiento de diferentes personas, pudiéndose tener subconjuntos de imágenes para cada persona que contengan diferentes posturas, condiciones de iluminación, etc.
- Crear una matriz formada por la nueva imagen de entrada y las ya almacenadas en la base de datos. Mediante un proceso matemático, se calculan los eigenvectores mediante la matriz de covarianza.

- Una vez obtenidos los vectores característicos, se comparan las distancias entre el vector que representa a la imagen original con el resto.
- Establecido un umbral de discernimiento a priori, si el menor valor del paso anterior es menor que este umbral, la imagen de la cara de entrada es considerada como conocida, si es mayor, se considera desconocida.



Imágen 3. Ejemplo de base de datos de caras. Fuente FERET ^[7]



Imágen 4. Eigenfaces de las caras de la Ilustración 3. Fuente: elaboración propia.

DESARROLLO

La implementación del proceso anteriormente mencionado se ha llevado a cabo utilizando la biblioteca libre de visión artificial OpenCV[8]. Es totalmente multiplataforma, existiendo versiones para Linux, Mac OS X y Windows, además contiene más de 500 funciones que abarcan una gran gama de áreas en el proceso de visión, como reconocimiento de objetos, reconocimiento facial, calibración de cámaras, visión estéreo y visión robótica.

Del desarrollo se han obtenido 2 aplicaciones:

- Aplicación servidor: multiplataforma, programada en lenguaje C haciendo uso de OpenCV, multihilo para que pueda atender peticiones de múltiples clientes simultáneamente.

```

Terminal — Detector — 80x24
Detector
MacBook-de-Juanvi-Martinez:Debug juanvimartinez$ ./Detector 5000
Server OnLine...
New client connected: 127.0.0.1
Cliente: 1      Recibido: n
Añadiendo nuevo archivo de imagen a persona existente
Image Size: 34591
Image Width: 120
Image height: 90

Obtenidas 7 caras de entrenamiento.
Cliente: 1      Recibido: stopReading
New client connected: 127.0.0.1
Cliente: 2      Recibido: r
Reconociendo persona en imagen...
Image Size: 34933
Image Width: 120
Image height: 90

Datos de entrenamiento cargados (7 imágenes de personas)
Mas cercano = 1, Confianza = 99.98%
Cliente: 2      Recibido: stopReading

```

Imagen 5. Servidor en ejecución y recibiendo peticiones. Fuente: elaboración propia.

- Aplicación cliente: multiplataforma, programada en JAVA haciendo uso también de OpenCV para facilitar el acceso a la cámara,



Imagen 6. Cliente en ejecución. Reconocimiento satisfactorio tras comunicarse con el servidor. Fuente: elaboración propia.

RESULTADOS

Tras las fases de investigación, estudio de requerimientos y desarrollo, se ha obtenido un sistema que cumple al 100% con las expectativas e ideas iniciales del proyecto.

Por una parte se ha desarrollado un software servidor para sistemas Linux que está continuamente escuchando en un puerto TCP a la espera de clientes y que, además, es el encargado de realizar todos los cálculos para determinar la identidad de una cara. Por otra parte, se ha desarrollado un software cliente de captura de imágenes en tiempo real que se encarga de realizar la extracción de la región de la cara deseada y comunicarse con el servidor para procesarla.

CONCLUSIONES

Recordando los objetivos iniciales del artículo “buscar la forma de integrar el reconocimiento facial en dispositivos con escasa capacidad de cálculo”, puede concluirse que dichos objetivos han sido cumplidos satisfactoriamente: se ha desarrollado un sistema de reconocimiento facial cliente-servidor robusto y rápido.

En el apartado resultados puede apreciarse como el reconocimiento facial en los casos con los que se ha experimentado ha sido correcto, obteniendo unas tasas de acierto cercanas al 100% y con unos tiempos de respuesta, tanto de la parte cliente como del servidor, más que aceptables.

No obstante, el sistema es susceptible a mejoras como son, por ejemplo, la fuerte dependencia de las condiciones de iluminación, supuestas constantes para el desarrollo del programa, y a la orientación y posición de la cara del sujeto, tanto en el proceso de detección como en el de reconocimiento.

REFERENCIAS

- [1] **R. and Poggio, T Brunelli**, *Face recognition: features versus templates*. IEEE Trans. PAM1, 1993, vol. 15.
- [2] **R. Chellappa, P.J. Phillips, A. Rosenfeld W. Zhao**, *Face Recognition: A literature survey.*: ACM Computing Surveys, 2003, vol. 35.
- [3] **G.H, Zahorjan, J Forman**, "The challenges of mobile computing," *IEEE*, vol. 27, no. 4, pp. 38-47, 1994.
- [4] **Paul V. and Michael J.** (2001) How face detection works. [Online]. http://www.cognotics.com/opencv/servo_2007_series/part_2/sidebar.html.
- [5] **R. and Maydt, J. Lienhart**, *An extended set of Haar-like features for rapid object detection.*: ICIP02, 2002.
- [6] **Matthew AT and Alex PP**, *Face recognition using eigenfaces.*: Proc. IEEE Conf. Computer Vision and Pattern Recognition, 1991.
- [7] Base de datos FERET. [Online]. http://es.wikipedia.org/wiki/Base_de_datos_FERET.
- [8] **OpenCV**. [Online]. <http://es.wikipedia.org/wiki/OpenCV>.