

Detección de Señas de Truco en Tiempo Real

Farré, Gonzalo. Salmerón, Facundo y Yackel, Francisco

Trabajo práctico final de materia: Procesamiento Digital de Imágenes – II - FICH-UNL.

Resumen— Se presenta en este trabajo una metodología para el reconocimiento de gestos faciales, aplicado a la detección de señas en el “Truco”, enfocando nuestro análisis en la mayoría de ellas. Para lograr este propósito, se procederá a la utilización de los contenidos aprendidos en la materia Procesamiento Digital de Imágenes, junto con librerías compatibles con el objetivo del proyecto.

Se realizaron pruebas pertinentes a la detección de rostros en tiempo real y mediante la utilización de ciertas métricas desarrolladas mediante la experimentación se pudieron obtener lineamientos para calcular las distintas señas que ejecutan los jugadores al momento de jugar “Truco”.

Concluimos que si bien logramos resultados gratificantes, nuestro análisis se podría extender a otros tipos de señas, en ambientes diversos.

Palabras clave— Truco, señas, reconocimiento, imagen, métricas.

I. INTRODUCCIÓN

El Truco, juego de naipes muy pupalar y reconocido en muchos países de sudamérica, consiste en emplear 3 cartas por jugador y combinarlas de manera tal de sacar la mayor cantidad de puntos por ronda. Terminando la partida al momento en que uno de ellos alcanza una cantidad de puntos determinada.

Este juego, puede ser llevado a cabo en una modalidad de dos equipos, donde cada uno de ellos puede ser de dos o más jugadores.

A partir de la necesidad de los jugadores de comunicar sus cartas a sus pares sin que los oponentes las reconozcan, el juego presenta un repertorio de muecas o gestos faciales predefinidos, los cuales varían dependiendo de la carta que el jugador posee.

Las diferentes señas utilizadas en el juego, varían desde las cartas de mayor jerarquía, a las de menor. A continuación, se le presentarán al lector las diferentes señas existentes:

- 1 de Espada (Macho): La seña consiste en levantar ambas cejas.



Figura 1: Seña correspondiente al 1 de Espada.

- 1 de Basto (Hembra): Esta seña consiste en guiñar un ojo.



Figura 2: Seña correspondiente al 1 de Basto

- 7 de Espadas: Esta seña se realiza desviando la boca hacia el lado derecho.



Figura 3: Seña correspondiente al 7 de Espadas

- 7 de Oro: Esta seña se realiza desviando la boca hacia el lado izquierdo.



Figura 4: Seña correspondiente al 7 de Oro

- 3: Esta seña se realiza mordiéndose el labio inferior de la boca.



Figura 5: Seña correspondiente al 3 de Espadas

- 2: Esta seña se realiza lanzando un beso.



Figura 6: Seña correspondiente a cualquier 2.

- 1 de Copa/ Oro: Esta seña se realiza abriendo la boca.



Figura 7: Seña correspondiente al 1 de Copa/Oro

- Cartas restantes: La seña se realiza cerrando ambos ojos.



Figura 8: Seña correspondiente a cualquier carta de baja jerarquía

En este análisis solamente nos enfocaremos en detectar todas las señas exceptuando las pertenecientes al 1 de basto, y a los 3.

II. PROCESAMIENTO DE LAS IMÁGENES

Como este proyecto se realizará estudiando los frames adquiridos en tiempo real, detallando los diferentes procesos por los que pasará la imagen, desde que se capta hasta obtener sus características, expresadas de manera tal de poder interpretarlas eficazmente, haciendo énfasis en las partes de interés.

Para lograr lo mencionado, se utilizó la librería de código abierto dlib junto con OpenCv. Gracias a ellas, se pudieron obtener las distintas regiones de interés dentro de las cuales se registran una serie de puntos (Landmarks) donde los mismos se distribuyen de manera uniforme a lo largo del rostro, por ejemplo detectando la boca, nariz, ojos y el contorno de la cara.



Figura 9: Marcadores faciales proveídos por biblioteca dlib

A. Adaptación de la seña a analizar

Este proceso se realiza luego de tomar imágenes preliminares, a partir de fotografías de las correspondientes señas las cuales nos servirán para poder identificar los diferentes puntos faciales de interés a la hora de realizar una mueca específica.

Para desarrollar estas pruebas, la biblioteca dlib provee un método del cual se pueden obtener las coordenadas de la imagen respecto a 68 puntos, distribuidos a lo largo de las distintas partes de la cara. En la siguiente imagen se puede visualizar la distribución de los mismos.

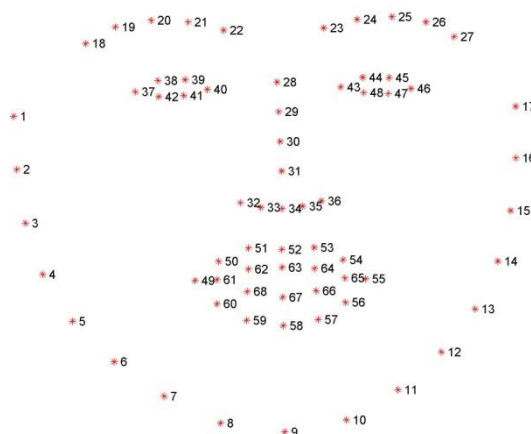


Figura 10: Distribución de los 68 marcadores faciales brindados por dlib

B. Prueba de concepto

Una vez definido el problema, y elegidas las bibliotecas para poder resolverlo, se realizó una prueba de concepto con el fin de poder comprobar de que la detección era posible.

Para la realización de la PoC se formó un conjunto de imágenes, donde las mismas mostraban las distintas señas realizadas por diferentes personas y además, una imagen de cada una de ellas con sus respectivas caras en expresión neutra.



Figura 11: Imágenes utilizadas para realizar PoC

Con el conjunto de datos, se prosiguió a buscar métricas sobre los distintos puntos que indican cuando se está haciendo una señal en particular. Para ello, se almacenaron las posiciones en la imagen de los mismos en su estado neutro y luego, se los tomó sobre la imagen que refleja la realización de una señal en particular. Por observación, se llegó a la conclusión de que el mejor estimador para medir los cambios entre las distintas imágenes neutras, y las que muestran una señal en particular, es la distancia Euclídea entre las coordenadas de interés dependiendo de la señal en cuestión.

Por ejemplo, para detectar cuando una persona está realizando la señal del “Macho”, las coordenadas de interés serán aquellas pertenecientes a la zona de los ojos y las cejas, por lo que un indicador de que se está realizando el gesto es un aumento de la distancia Euclídea entre los marcadores de la imagen neutra, y la imagen de la señal.

Después de probar y encontrar para todas las señales cuales eran los indicadores que mejor aproximan la solución, se decidió dar por terminada la PoC para poder dar paso a la implementación y desarrollo de la aplicación en tiempo real.

C. Adaptación del ambiente y las herramientas de trabajo.

Para poder desarrollar la aplicación y que la misma pueda detectar las señales en tiempo real, se tuvo que contar con el apoyo y la utilización de la biblioteca OpenCv, la cual se encarga de brindar métodos para poder obtener cada frame que va registrando la cámara.

Una vez que se obtuvo el frame, al mismo hay que convertirlo a un objeto de dlib ya que ésta será la encargada de realizar el procesamiento y detección sobre la imagen.

III. DESARROLLO DE LA APLICACIÓN

Para que la detección pueda realizarse en tiempo real, se deben guardar aquellos puntos a utilizar correspondientes a 20 frames anteriores al actual, a fin de poder tener un objeto de comparación. Es por ello, que el usuario no debe lanzar la aplicación realizando señales ya que el sistema será incapaz de poder detectarlas desde un principio.

Como punto de partida del desarrollo, se necesitó de un detector de caras, el cual provee los puntos de interés para el frame de estudio, por lo que se utilizó un método perteneciente a dlib el cual está implementado con el fin de registrar caras frontales. Después de esto, lo que hacemos es convertir el frame obtenido a través de OpenCv a un objeto perteneciente a dlib sobre el cual se realizará el procesamiento.

Siguiendo con el desarrollo, y ya contando con la imagen lista para manipular, se utiliza la opción de dlib por la cual se obtienen los 68 puntos de interés correspondientes a la cara.

A partir de aquí, ya se cuenta con valores discretizados los cuales serán evaluados con aquellos que fueron guardados con anterioridad. Se calculan las diferentes métricas de interés correspondientes a todas las señales posibles a realizar por el usuario.

Por último, sólo resta preguntar en diferentes condicionales que situación se está llevando a cabo e

informar al usuario la señal que se está realizando. Recordar que a medida que van pasando los frames, cada 20 de ellos se actualiza aquel frame denominado neutral.

IV. RESULTADOS

Los datos obtenidos de las pruebas se han volcado en una tabla para poder observar los porcentajes de acierto por cada una de las señales, utilizando 10 personas como modelos de pruebas.

Señal	Porcentaje de acierto
Macho	90%
7 de Espada	75%
7 de Oro	60%
2 de cualquier tipo	50%
1 de Oro/Copa	75%
Ninguna carta de renombre	100%
Total	75%

Figura 12: Resultados de pruebas con 10 personas

Como se puede observar se lograron resultados confiables tanto en la señal del Macho, como en la señal que indica que no se posee ninguna carta significativa. No tan así en la señal del 2 como en la de los distintos 7s. Se piensa que esto puede deberse a la heterogeneidad de señales que se realizan utilizando la boca.

V. CONCLUSIONES

Se desarrolló una técnica de detección y reconocimiento de las señales faciales utilizadas en el juego “Truco”, basándonos en la utilización de marcadores provistos por la biblioteca dlib junto con la utilización de métodos importados de OpenCv. Dicha técnica fue validada al probar con distintas señales provenientes de diferentes personas, donde se verificó que en la mayoría de las situaciones el resultado fue satisfactorio.

Al analizar la tabla que se encuentra en la sección resultados, se puede verificar que la aplicación diseñada tiene un 75% de efectividad.

Es importante destacar que la metodología para el cálculo de las métricas utilizadas para definir cuál señal se está haciendo fue el cálculo de la distancia Euclídea entre un mismo punto en diferentes frames.

Este trabajo podría extenderse evaluando como encontrar la forma de detectar las señales que no fueron incluidas en el alcance del proyecto. Además, podría definirse un protocolo de pruebas a fin de poder entrenar una red neuronal ajustando sus parámetros e indicando automáticamente que señal se está haciendo.

VI. BIBLIOGRAFÍA

- [1] Hsu, R. L., Abdel-Mottaleb, M., & Jain, A. K. (2002). Face detection in color images. *IEEE transactions on pattern analysis and machine intelligence*, 24(5)
- [2] Zhu, X., & Ramanan, D. (2012, June). Face detection, pose estimation, and landmark localization in the wild. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on* (pp. 2879-2886).
- [3] Segundo, M. P., Silva, L., Bellon, O. R. P., & Queirolo, C. C. (2010). Automatic face segmentation and facial landmark detection in range images. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 40(5), 1319-1330.
- [4] Rafael C. González and Richard E. Woods, *Digital Image Processing*, Prentice-Hall, 3rd. ed. (2008)