Reconocimiento de emociones a través de expresiones faciales con el empleo de aprendizaje supervisado aplicando regresión logística.

Carlos Barrionuevo, Jorge Ierache, Iris Sattolo
Instituto de Sistemas Inteligentes y Enseñanza Experimental de la Robótica. SECYT
Escuela Superior de Ingeniería, Informática y Ciencias Agroalimentarias
Universidad de Morón Cabildo 134, Buenos Aires, Argentina
{cbarrionuevo,jierache,isattolo}@unimoron.edu.ar

Abstract. A través de las expresiones faciales se trasmite más de la mitad del significado de un mensaje. Diferentes estudios, han demostrado además que algunas de estas tienen el carácter de universales. Sin embargo, una de las características más interesantes de las expresiones faciales es que revelan emociones. Esta cualidad, ha hecho que diversas disciplinas se hayan volcado a su estudio con diferentes objetivos, la computación no fue la excepción. La detección de rostros y sus partes principales, ha sido uno de los grandes avances en el área de la visión por computadora. Esto, sumado al auge de la última década del aprendizaje automático, posibilitó el desarrollo de sistemas capaces de detectar emociones a través del análisis de expresiones faciales. En este trabajo describiremos las diferentes etapas del desarrollo, entrenamiento y prueba de un algoritmo de regresión logística para la detección de emociones.

Keywords: Expresiones faciales, emociones, visión por computadora, aprendizaje automático, regresión logística**.**

1 Introducción

A partir del nuevo siglo, mejorar la interacción entre el ser humano y las máquinas se ha convertido en uno de los principales objetivos de la computación. Con este fin, diferentes líneas de investigación se centran en crear nuevas formas de interactuar. La computación afectiva [1] consiste en brindarles a las computadoras la capacidad de interpretar el estado emocional del usuario. El análisis de las expresiones faciales constituye una de las formas más eficaces para revelar el estado emocional de un individuo. Albert Mehrabian en sus investigaciones afirma que en una comunicación la palabra hablada solo contiene el 7% del significado, mientras que el tono de voz un 38% y las expresiones faciales nada menos que el restante 55% [2]. El presente trabajo tuvo como objetivo describir el proceso de construcción de un sistema de detección de emociones capaz de reconocer las 7 expresiones emocionales catalogadas como universales por la teoría de Paul Ekman. En la sección 2 se describe la base de datos de imágenes de rostros utilizada, y se detalla el proceso de extracción de las características necesarias para entrenar un algoritmo de aprendizaje automático.

En la sección 3 se presenta el algoritmo utilizado mientras que en la sección 4 se muestran los experimentos realizados con el algoritmo implementado en los que se compara su rendimiento con otro servicio de similares características, y por último en la sección 5 se brindan las conclusiones y futuras líneas de trabajo.

2 Enfoque categórico- Base de datos de imágenes de rostros

En el marco del enfoque categórico, Paul Ekman plantea en su teoría la existencia de seis expresiones faciales universales que trascienden el idioma y las diferencias regionales, culturales y étnicas; a las que relaciona con seis emociones basales: enojo, asco, felicidad, miedo, tristeza y sorpresa (en inglés “anger”, “disgust”, “fear”, “happiness”, “sadness” y “surprise”). Posteriormente en su trabajo [3] Ekman adiciona una séptima expresión facial que representa la emoción “desprecio” (en inglés “contempt”). En la figura 1a se muestran 7 imágenes representativas de las 7 expresiones faciales universales tomadas de la página web de Paul Ekman [4] con el titulado de la foto entre paréntesis Para la presente investigación se utilizó la base de datos de rostros “RaFD”[5] desarrollada por la Universidad de Radbound de la ciudad de Nijmegen (Holanda). Esta se compone de imágenes de rostros de 67 modelos en su mayoría de raza caucásica, de ambos sexos, adultos y niños. De acuerdo con el Sistema de codificación facial [6] (Facial Action Coding System, “FACS”) desarrollado por Paul Ekman, cada modelo aparece en el set fotografiado desde cinco ángulos distintos (0°,45°,90°,135° y 180°) y las imágenes frontales (90°) tomadas con el sujeto dirigiendo la mirada hacia 3 direcciones diferentes (izquierda, al frente y hacia la derecha). Este set considera las 7 expresiones faciales universales definidas anteriormente y adiciona una octava expresión catalogada como “neutral”. Todas estas características lo hacen apto para múltiples campos de investigación tales como señales faciales de atención y procesamiento de expresiones faciales. Si bien existen otras bases de datos de rostros tales como “Jaffe” [7] también conocida como “la base de imágenes de mujeres japonesas” y el “Cohn-Kanade dataset” [8], se eligió “RaFD” por las siguientes razones: a) Alta Calidad y resolución de las imágenes (1024 x 681 pixeles). Las sesiones de fotos se desarrollaron en un ambiente altamente controlado, con condiciones óptimas de luz. b) Variedad de características de los modelos tales como raza, género y edades, c) Cada una de las imágenes se encuentra debidamente etiquetada con la expresión facial emocional mostrada en ella a través del nombre del archivo, d) Sesión de fotos dirigida y asistida por especialistas certificados en el Sistema de codificación facial. Si bien esta característica es compartida entre diversos sets de imágenes, no es menor y aporta confiabilidad para la investigación. Dentro del nombre del archivo de cada una de las imágenes se puede encontrar el nombre de la base de datos, el ángulo de la toma, el número de modelo, la raza, el género, la emoción mostrada y la dirección de la mirada. Por ejemplo, si el nombre de la imagen es el siguiente “Rafd090_07_Caucasian_male_sad_left”, indica que la imagen fue tomada a 90 grados (de frente), el número de modelo es el “07”, la raza del mismo es caucásica, el género es masculino, la emoción mostrada es “tristeza” (en idioma inglés “sad”) y la mirada del sujeto apunta hacia la izquierda (en idioma inglés “left”).

Como el set incluye imágenes tomadas desde cinco ángulos diferentes solo se utilizaron las tomadas a noventa grados (de frente) obteniendo un total de 460 imágenes. De este subconjunto se reservó un 90% de las mismas (414 imágenes) para el entrenamiento [9] del algoritmo de regresión logística que se aplicó en el reconocimiento de emociones. El restante 10% (46 imágenes) se guardó para la realización de pruebas y experimentos de predicción [9]. Esta última parte se eligió de forma tal que se eliminó por completo del conjunto de imágenes que se utilizaron para entrenamiento a los modelos número 16,18,23,24,25,30,32,36,38 y 54.

Presentada la base de datos de rostros con la que se trabajó, el siguiente paso fue desarrollar una rutina que posibilite la extracción de las características necesarias de cada imagen del set. El lenguaje de programación elegido fue "Python", ya que cuenta con la biblioteca "dlib" [10] que proporciona las herramientas necesarias para el desarrollo. La primera tarea, consistió en obtener del nombre del archivo la etiqueta de la emoción asociada al mismo. En base a esta se asoció un código numérico que consistió en un valor entre uno y ocho. De esta manera, se representó de forma univoca cada emoción. En la tabla 1 se detalla la codificación de emociones utilizada:

Tabla 1. Codificación de emociones.

Etiqueta	Traducción	Código
Angry	Enojo	1
Contemptuous	Desprecio	2
Disgusted	Asco	3
Fearful	Miedo	4
Happy	Feliz	5
Neutral	Neutral	6
Sad	Tristeza	7
Surprise	Sorpresa	8

Como resultado de esta tarea se obtuvo un vector identificado con la letra "Y" de "K" dimensiones, donde "K" es el número de imágenes del set que se utilizaron para entrenar el algoritmo de regresión logística. El vector final, fue almacenado en un archivo con extensión ".txt". El siguiente paso fue para cada una de las imágenes, detectar la región del rostro (también conocida como "región de interés"). Esto permitió poder identificar sobre esta una serie de puntos característicos (también conocidos como "landmarks"). Existen una gran variedad de detectores de puntos característicos del rostro que varían (entre otras cosas) en la cantidad de "landmarks" que identifican; pero coinciden generalmente en la localización de las regiones de la boca, cejas, nariz, ojos y contorno de la cara. Para este trabajo, se utilizó el detector incluido en la biblioteca "dlib" [10]. El mismo es una implementación del algoritmo desarrollado por Vahid Kazemi y Josephine Sullivan [11]. Este permite obtener las coordenadas en los ejes "x" e "y" de 68 puntos característicos de la cara. La localización de estos se puede observar en la figura 1b [12]. Una vez localizados los landmarks, considerando los pares (x,y) de cada uno se totalizan 136 elementos con los cuales se generó una fila por imagen. De esta forma las características faciales de cada una de las imágenes se vieron representadas por su correspondiente tupla. De forma general, el resultado de este proceso es una matriz de "m" filas y "n" columnas. Siendo "m" el número de ejemplos de entrenamiento y "n" el número de

características. Para el presente caso de estudió, “m” fue igual a 414 (número de imágenes que se utilizó para entrenamiento del algoritmo) y “n” como se acaba de mencionar a 136. Esta matriz se identificó con la letra “X” y se conoce como “matriz de características”. Al igual que el vector “Y” se almacenó en un archivo con extensión “txt”. Una vez descriptas ambas matrices, resulta importante mencionar que ambas fueron generadas en simultaneo, por lo que la fila “i” de la matriz “X” se correspondió con el elemento “i” del vector “Y”. Finalizado este proceso, ya se cuenta con la entrada necesaria para el entrenamiento del algoritmo de aprendizaje supervisado.

Fig. 1a Siete Expresiones faciales universales según la teoría de Paul Ekman.

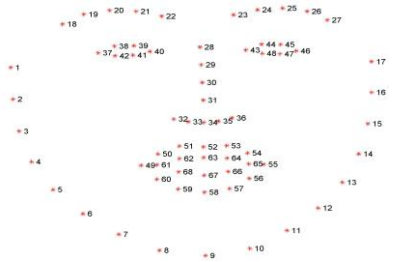

Fig. 1b Distribución de landmarks en el rostro (fuente pyimagesearch).

3 Predicción de emociones con el empleo de Regresión Logística.

Dada una nueva imagen del rostro de una persona, para predecir cual es la emoción que está expresando, una de las posibilidades es utilizar un algoritmo de aprendizaje supervisado. Se le dice “supervisado” ya que para su entrenamiento necesita de un conjunto de datos previamente etiquetado y clasificado. Sobre este grupo de datos conocido como “conjunto de datos de entrenamiento” el algoritmo realizará predicciones y las comparará con las etiquetas, con el error obtenido y a través de sucesivas iteraciones irá ajustando el modelo logrando así un aprendizaje progresivo. Existen una gran variedad de algoritmos de estas características tales como la regresión lineal, regresión logística, redes neuronales, máquina de soporte de vectores, K vecinos más próximos, etc [13]. La utilización de uno u otro generalmente depende de las dimensiones del problema [14]. Para este trabajo se optó por un algoritmo de regresión logística [15]. La implementación del mismo se llevó a cabo en el lenguaje de programación Octave [16]. Este algoritmo de clasificación nos permite a través de un clasificador estimar la probabilidad de que un nuevo ejemplo pertenezca a una clase. Como para el presente problema se tuvo un total de ocho clases, se aplicó además una técnica denominada “one vs all” [14] (uno contra todos). De esta forma, con la ayuda de este método fue posible entrenar ocho clasificadores. Uno por cada una de las emociones consideradas. Cada clasificador se ve representado por su correspondiente función hipótesis. Primero, para simplificar la explicación definiremos la función polinómica “z” (que forma parte de la hipótesis) según la propuesta de Andrew Ng [14]:

$$z(x) = \theta^T.X \qquad (1)$$

Donde:

- “θ” es una matriz de “i” filas y “j” columnas. Siendo “i” el número de clases y “j” el número de parámetros del polinomio. Para el caso en estudio, “i” será igual a ocho (número de emociones) y “j” equivalente al número de características (136) que conforman los “landmarks”. Inicialmente, todos los elementos de esta matriz fueron iguales a 0 (cero). Estos son los parámetros que se ajustaron a través del entrenamiento para luego poder realizar predicciones.
- “X” es la matriz de características definida en el apartado anterior.

Definida la función “z”, la expresión de la función hipótesis para regresión logística [14] será la siguiente:

$$h_\theta^i(x) = \frac{1}{(1 - e^{(-z)})} \quad (\mathbf{2})$$

Donde:

- “i” se refiere a la i-ésima clase.

La función anterior consiste en aplicar la función sigmoide a “z(x)”. Esto produce que el resultado de esta sea un valor entre 0 y 1. Este representa la probabilidad de que un nuevo ejemplo de entrenamiento (para este caso las características de un nuevo rostro) pertenezca a la clase (emoción) “i”. La etiqueta de la emoción del clasificador que arroje la mayor probabilidad será la salida final del algoritmo.

Por otra parte, Se denomina “costo” a la penalización que pagará el algoritmo dado un valor de probabilidad calculado por la función hipótesis en caso de que la etiqueta sea “y”. El costo para regresión logística se define a través de la siguiente expresión [14]:

$$costo(h_\theta(x), y) = -y.\log\big(h_\theta(x)\big) - (1 - y).\log(1 - h_\theta(x)) \quad (\mathbf{3})$$

Si y = 1 (estimamos la probabilidad de que pertenezca a esa clase), la representación del costo se observa en la figura número 2. Por ejemplo, si dada una nueva imagen que muestra la emoción “enojo” (y=1) pueden darse las siguientes situaciones:

- Si el valor calculado por la función hipótesis h(x) asociada al clasificador de la emoción “enojo” calcula un valor de 1 (100% de probabilidad de que la emoción mostrada sea “enojo”), el costo que pagará el algoritmo será 0.
- Si el valor calculado por la función hipótesis h(x) asociada al clasificador de la emoción “enojo” tiende a 0 (0% de probabilidad de que la emoción mostrada en la imagen sea “enojo”) el costo que pagará el algoritmo tenderá a infinito.

Definido el costo, la expresión de la función de costos es la siguiente [14]:

$$J(\theta) = -\frac{1}{m}\sum_{i=1}^{m}[y^{(i)}\log(h_\theta(x^{(i)}) + (1 - y^{(i)})\log(1 - h_\theta(x^{(i)}))] \quad (4)$$

Donde:

- “m” es igual al número de ejemplos de entrenamiento. (en nuestro caso 414 instancias de rostros)

Como es posible observar, en la expresión (4), la función de costos calcula el error promedio entre todos los ejemplos del set de entrenamiento.

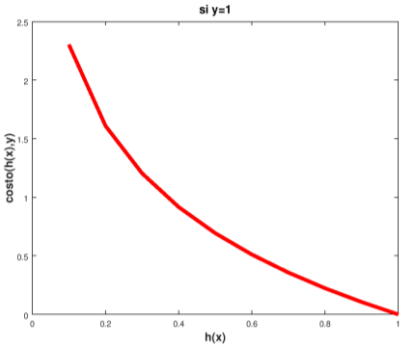

Fig. 2 Representación del costo.

Respecto de la matriz "θ", tal como se mencionó antes, inicialmente todos sus elementos serán iguales a 0(cero). El objetivo es entonces encontrar aquellos parámetros “θ" que minimicen el error de la función de costos. Para esta tarea existen diferentes técnicas, tales como el descenso de gradiente cuya implementación fue considerada para este trabajo, pero finalmente se optó por una función de optimización avanzada provista por "Octave" denominada "fminunc" ("function minimization unconstrained") [17]. “fminunc” recibe como entradas un puntero a una función “f”, una matriz de parámetros “x” y un vector de opciones clave-valor adicionales entre las que es importante mencionar el número máximo de iteraciones (clave “MaxIter”). Esta función retorna la matriz de parámetros “x” de forma tal que “f(x)” es un mínimo local [18]. Para el presente trabajo se parametrizó dicha función con: a) un puntero a la función costos, b) la matriz de parámetros “θ” y c) la clave “MaxIter” definida con el valor 1200 que indica la cantidad de veces que se iteró sobre el set de datos buscando converger en un mínimo local de la función de costos. La salida de esta función es la matriz de parámetros ajustada en base a nuestros datos de entrenamiento.

4 Experimentos y resultados obtenidos

El set de pruebas que se utilizó, como se mencionó al momento de describir la base de datos de rostros, se compone de 46 imágenes, las cuales corresponden a los sujetos (modelos) 16,18,23,24,25,30,32,36,38 y 54. En la tabla 2 se muestra la descripción del set de pruebas:

Tabla 2. Descripción del set de pruebas.

Código	Descripción	Cantidad de imágenes	Modelos utilizados
1	Enojo (angry)	5	16,18,30,32 y 36.
2	Desprecio (contemptous)	7	16(2),18,23,25,30 y 32.
3	Asco (disgusted)	7	18(2),25,30(2),32 y 38.
4	Miedo (fearful)	8	16(2),18,25(3),30 y 32.
5	Felicidad (happy)	4	24,25,30 y 54.
6	Neutral (neutral)	4	24,25,30 y 32.
7	Tristeza (sad)	5	16,18,25 y 30(2).
8	Sorpresa (surprised)	6	16,25,30,32,36 y 38.

En la tabla 2 en la primera columna se indica el código de emoción utilizado en el presente trabajo (de acuerdo con lo definido en la tabla 1), en la segunda columna la descripción de dichos códigos (nombre de la emoción), en la tercer columna se encuentra la cantidad de imágenes presentes en el set de pruebas relacionadas a cada emoción y en la última columna “Modelos utilizados” se indica el número de modelo y entre paréntesis la cantidad de imágenes del mismo (si fuera más de una).Al igual que en la etapa de entrenamiento, fue necesario generar la matriz de características para el set de datos de pruebas. La misma fue de 46 filas (número de imágenes) y 136 columnas (cantidad de características). La matriz resultado, fue almacenada también en un archivo con extensión "txt". Esta tarea es realizada por una rutina desarrollada en lenguaje Python, como se indicó anteriormente. Una vez obtenida la matriz "X", el siguiente paso fue evaluar las 136 características de cada una de las imágenes en las funciones hipótesis correspondientes a los 8 clasificadores emocionales. El resultado de esta operación fue una matriz de 8 filas y 46 columnas. La cantidad de filas se corresponde con la cantidad de clasificadores (de las distintas emociones), el número de fila indica el código de emoción de acuerdo con lo definido en la tabla 1. De esta forma, en la fila 1 se encontrarán las salidas del clasificador asociado a la emoción con código 1 (Enojo). La cantidad de columnas por su parte es igual al número de imágenes del set de pruebas. Así, por ejemplo, en la columna 1 se obtuvieron las salidas de los 8 clasificadores para la primera imagen. La salida final del algoritmo es para cada imagen el número de fila (que como acabamos de mencionar se corresponde con el número de clasificador emocional) para el que se calculó el máximo valor de probabilidad. Los resultados obtenidos posibilitaron la construcción de la matriz de confusión [19] de la figura 3:

		Esperado								Precisión
		Enojo	Desprecio	Asco	Miedo	Felicidad	Neutral	Tristeza	Sorpresa	
		1	2	3	4	5	6	7	8	
Predecido	1	5	0	0	0	0	1	1	0	0,83
	2	0	4	0	0	0	1	0	0	0,8
	3	0	0	7	0	0	0	0	0	1
	4	0	0	0	7	0	0	0	0	1
	5	0	0	0	0	4	0	0	0	1
	6	0	1	0	0	0	2	0	0	0,67
	7	0	2	0	1	0	0	4	0	0,57
	8	0	0	0	0	0	0	0	6	1
Recall		1	0,57	1	0,88	1	0,5	0,8	1	85%

Fig. 3 Matriz de confusión construida con los resultados obtenidos para la prueba con 46 imágenes.

De la matriz presentada en la figura 3 y en función de la cantidad de imágenes por emoción indicadas en la tabla 2, es posible deducir que el sistema desarrollado en el presente trabajo falló en: a) 3 imágenes donde el resultado esperado era “desprecio” (código 2) y el valor predicho fue en 2 ocasiones la emoción “tristeza” (código 7) y en la restante la emoción “neutral” (código 6), acertando en las cuatro restantes imágenes correspondiente a desprecio, para un total de 7 imágenes (desprecio); b) en una imagen donde el resultado esperado era “miedo” (código 4) el valor predicho fue “tristeza” (código 7) para un total de 8 imágenes (miedo); c) en 2 imágenes en las cuales se esperaba como resultado “neutral” (código 6) el algoritmo las confundió con las emociones “enojo” (código 6) y “desprecio” (código 2), para un total de 4 imágenes (Neutral); d) en una imagen en donde el resultado esperado era “tristeza”

(código 7) el valor predicho fue “enojo” (código 1) para un total de 5 imágenes de tristeza . Totalizando 7 predicciones erróneas y 39 predicciones correctas. Además, con las predicciones realizadas se calcularon 3 métricas diferentes que nos permitieron evaluar el rendimiento del modelo predictivo propuesto (algoritmo de regresión logística). La primera de ellas se denomina “Recall” [20] y mide para cada una de las clases de emociones que fracción del total de imágenes disponibles para esa clase fueron correctamente predichas. Tomando como ejemplo la emoción “Desprecio” y observando la columna relacionada a la misma en la figura 3, notamos que de 7 imágenes disponibles para dicha emoción el algoritmo acertó en la predicción de 4, obteniendo un “recall” de 0,57 (57%) para dicha emoción. La segunda métrica que se calculó fue la “Precisión”, esta indica para cada uno de los 8 clasificadores emocionales, de la cantidad de predicciones realizadas por estos cuantas fueron correctas. Observando por ejemplo la fila relacionada a la emoción “Tristeza” (código 7) en la figura 4, se nota que de 7 ocasiones en las que el valor predicho fue esta emoción, se acertó en 4, obteniendo una precisión de 0,57 (57%). Finalmente, la última métrica que se calculó fue la llamada “Accuracy” [20] (exactitud). Esta a diferencia de las dos métricas anteriores mide el desempeño global del algoritmo y no el particular de cada emoción. Se calcula como el total de predicciones realizadas correctamente sobre el total de predicciones realizadas. Como se comentó anteriormente el total de predicciones correctas en esta prueba fue de 39 sobre 46 predicciones totales, obteniendo un “accuracy” del 85%. Realizados los testeos con el set de datos, se llevó a cabo además una prueba independiente con imágenes externas al mismo, en este caso fueron seleccionadas las imágenes presentadas en la figura 1a que como se mencionó antes, fueron tomadas de la página web de Paul Ekman [4] y consideran distintos sujetos para la representación de las distintas emociones. Para esta prueba, las imágenes fueron clasificadas con el sistema propuesto y se realizó una comparación con el servicio de inteligencia artificial “Face” de Microsoft [21], con independencia de la cantidad de landmarks de cada modelo, el cual también considera las mismas ocho emociones planteadas para este trabajo. Los resultados obtenidos se muestran en la figura 4, donde es posible observar los valores de probabilidad calculados por los 8 clasificadores emocionales de ambos sistemas para cada una de las imágenes. se puede observar que el sistema de Microsoft Face falló en la predicción de la emoción “desprecio” mientras que el modelo predictivo desarrollado lo hizo en las emociones “enojo” y “miedo”. En la figura 4, en la primera fila se muestran los títulos de las imágenes y entre paréntesis se aclara el código de emoción asociado al resultado esperado. Para cada imagen de la primera fila, se detallan los resultados obtenidos por el servicio “Face” y por el sistema desarrollado propuesto en este articulo (S.P.). La primera columna por su parte muestra el código de emoción seguido por el nombre de esta presentado en la tabla 1 que fue con el que se trabajó en este artículo. La salida final del algoritmo de regresión logística que se construyó en este trabajo fue la que se muestra en la figura 5. Como segunda prueba independiente, se tomó uno de los sujetos del set de pruebas inicial compuestos por 46 fotografías de la base de datos de rostros “RaFD”[5]. El modelo elegido fue el número 30 [9], ya que se contaba con al menos una imagen de cada emoción (8 imágenes en total), y se realizó nuevamente una comparación con el servicio de Microsoft. Los resultados obtenidos se muestran en la figura 6.

	(1) anger		(2) contempt		(3) disgust		(4) fear		(5) happiness		(7) sadness		(8) surprise	
	Face	S.P.	Face	S.P.	Face	S.P.	Face	S.P.	Face	S.P.	Face	S.P.	Face	S.P.
1 - Enojo	**0,472**	0	0	0	0,006	0	0	0	0	0	0	0	0	0
2 - Desprecio	0,006	0	0,213	**0,01**	0	0	0	0	0	0,012	0	0,018	0	0
3 - Asco	0,001	0	0	0	**0,993**	**0,043**	0	0	0	0	0	0	0	0,02
4 - Miedo	0,038	0	0	0	0	0	**0,991**	0,153	0	0	0	0,001	0,006	0
5 - Feliz	0	0	0,004	0	0	0	0	0,001	**1**	**0,465**	0	0	0	0
6 - Neutral	0,089	**0,056**	**0,772**	0,007	0	0	0	0,071	0	0	0,124	0,005	0,017	0
7 - Tristeza	0,001	0,002	0,011	0	0	0	0,001	**0,916**	0	0	**0,876**	**0,762**	0	0
8 - Sorpresa	0,392	0	0	0	0	0,002	0,008	0	0	0	0	0	**0,977**	**0,124**

Fig. 4 Resultados obtenidos entre el sistema propuesto y el servicio “Face” de Microsoft.

```
Resultado:

Ordinal (1) Resultado: 6 - Neutral
Ordinal (2) Resultado: 2 - Desprecio
Ordinal (3) Resultado: 3 - Asco
Ordinal (4) Resultado: 7 - Tristeza
Ordinal (5) Resultado: 5 - Felicidad
Ordinal (6) Resultado: 7 - Tristeza
Ordinal (7) Resultado: 8 - Sorpresa
```

Fig. 5 Salida final del sistema propuesto para la prueba con el set de imágenes de la página web de Paul Ekman.

	(1) angry		(2) contemptous		(3) disgusted		(4) fearful		(5) happy		(6) neutral		(7) sad		(8) surprised	
	Face	S.P.	Face	S.P.	Face	S.P.	Face	S.P.	Face	S.P.	Face	S.P	Face	S.P.	Face	S.P.
1 - Enojo	0,291	**1**	0	0	0,66	0,005	0	0	0	0	0	0,074	0,006	0	0	0
2 - Desprecio	0,004	0,931	0,002	0,046	0,001	0	0,014	0,001	0	0	0	0,05	0,094	0	0	0
3 - Asco	0	0	0	0	**0,339**	**0,999**	0,225	0	0	0,085	0	0	0,013	0	0	0
4 - Miedo	0	0	0	0	0	0	**0,679**	**1**	0	0	0	0	0,016	0,354	0	0
5 - Feliz	0	0	0	0	0	0	0	0	**1**	**1**	**0**	0	0	0	0	0
6 - Neutral	**0,704**	0	**0,998**	**0,957**	0	0,004	0	0	0	0	**1**	**0,993**	0	0,1	0	0
7 - Tristeza	0	0	0	0	0	0	0,007	0,713	0	0	0	0	**0,644**	**0,998**	0	0,137
8 - Sorpresa	0,001	0	0	0	0	0	0,074	0,002	0	0	0	0	0,026	0	**1**	**1**

Fig 6 Resultados obtenidos de la comparación entre el sistema propuesto y el servicio “Face” de Microsoft para la prueba con el sujeto número 30.

Es posible observar en la figura 6 que tanto el sistema desarrollado para el presente trabajo como el servicio de Microsoft fallaron en la imagen que expresaba “desprecio” calculando ambos sistemas el mayor valor de probabilidad en la emoción “neutral” (99,8% calculado por el servicio “Face” y 95,7% calculado por nuestro sistema). Además, el servicio de Microsoft erró en la predicción de la fotografía que expresaba “enojo” confundiéndola también con la expresión “neutral”.

5 Conclusión y futuras líneas de trabajo

De acuerdo con el análisis realizado sobre los resultados obtenidos, podemos concluir que el sistema construido presentó un desempeño correcto tanto en las pruebas con imágenes del set de entrenamiento como con imágenes externas al mismo. Existe un margen de mejora que podría darse entrenando al algoritmo con un número mayor de imágenes, con modelos con mayor diversidad de características, o modificando el proceso de extracción de características, obteniendo por ejemplo el grado de apertura de los ojos, boca, etc. En el marco de futuras líneas de trabajo se evaluará el rendimiento de otras técnicas de aprendizaje supervisado (tales como redes neuronales o máquina de soporte de vectores) en la tarea de detección de emociones y elaborar una comparación con la aplicación construida en la presente investigación. Además, se contemplará la integración en sistemas multimodales [22] [23] combinado

diferentes sensores (variación de ritmo cardiaco, conductancia de piel, eeg a través de interfaces cerebro-máquina), reforzando de esta forma la determinación categórica de emociones realizada a través de la detección de rostros. Esto permitirá la integración en diversos dominios de aplicación en el marco de la computación afectiva.

6 Referencias

1. Picard, R.W., et al.: Affective Computing (1995)
2. A. Mehrabian, “Communication without Words”, PsychologyToday,vol. 2, no 4, (68)
3. Ekman, P., Friesen, W. V., O'Sullivan, M., Chan, A., Diacoyanni-Tarlatzis, I., Heider, K., A. (1987). Universals and cultural differences in the judgments of facial expressions of emotion. *Journal of Personality and Social Psychology, 53*(4), 712–717.
4. Paul Ekman Group, https://www.paulekman.com/. Accedido en junio 2020.
5. Langner, O., Dotsch, R., Bijlstra, G., Wigboldus, D.H.J., Hawk, S.T., & van Knippenberg, A. (2010). Presentation and validation of the Radboud Faces Database.Cognition & Emotion, 24(8), 1377—1388. DOI: 10.1080/02699930903485076
6. Ekman, P., Friesen, W. V., & Hager, J. C. (2002a). Facial Action Coding System: The manual. Salt Lake City, UT: Research Nexus
7. Michael J. Lyons, Shigeru Akamatsu, Miyuki Kamachi, Jiro Gyoba.Coding Facial Expressions with Gabor Wavelets, 3rd IEEE International Conference on Automatic Face and Gesture Recognition, pp. 200-205 (1998).
8. Lucey, P., Cohn, J. F., Kanade, T., Saragih, J., Ambadar, Z., & Matthews, I. (2010). The Extended Cohn-Kanade Dataset (CK+): A complete expression dataset for action unit and emotion-specified expression.
9. Conjuntos de imágenes de rostros usados para entrenamiento y pruebas, https://drive.google.com/drive/folders/1AhYfPoBoBp0oU4WBWgSmXeD8xTPKqXLJ?usp=sharing.
10. Dlib, https://pypi.org/project/dlib/
11. Kazemi and J. Sullivan. One millisecond face alignment with an ensemble of regression trees, 2014
12. Facial landmarks with dlib, OpenCV, and Python. Pyimagesearch https://www.pyimagesearch.com/2017/04/03/facial-landmarks-dlib-opencv-python/
13. Tipos ML ,https://medium.com/soldai/tipos-de-aprendizaje-autom%C3%A1tico-6413e3c615e2 .
14. Andrew Ng. Machine Learning. Coursera. https://www.coursera.org/learn/machine-learning
15. La Regresión logística, https://www.analyticslane.com/2018/07/23/la-regresion-logistica/.
16. Octave, https://www.gnu.org/software/octave/.
17. Fminunc, https://octave.sourceforge.io/octave/function/fminunc. Accedido en junio 2020.
18. Minimizers, http://octave.org/doc/v4.4.1/Minimizers.html. Accedido en septiembre 2020.
19. Multi-Class Metrics Made Simple, Part I: Precision and Recall. Towards data science. https://towardsdatascience.com/multi-class-metrics-made-simple-part-i-precision-and-recall-9250280bddc2. Accedido en junio 2020.
20. Confusion Matrix For Your Multi-Class Machine Learning Model. Towards data science. https://towardsdatascience.com/confusion-matrix-for-your-multi-class-machine-learning-model-ff9aa3bf7826. Accedido en junio 2020.
21. Face, Microsoft. https://azure.microsoft.com/es-mx/services/cognitive-services/face/. Junio 2020.
22. Ierache, J., Nicolosi, R., Ponce, G., Cervino, C., & Eszter, E. (2018). Registro emocional de personas interactuando en contextos de entornos virtuales. XXIV CACIC 2018, (págs. 877-886).
23. Ierache, J., Ponce, G., Nicolosi, R., Sattolo, I., & Chapperón, G. (2019). Valoración del grado de atención en contextos áulicos con el empleo de interfase cerebro-computadora. CACIC 2019, Libro de actas pp 417-426