

# Proyecto Final

## “Detección de Sentimiento en video.”

Manuel Borja Aquino, Daniela Niño de Rivera, Arturo Tellez Cortez and Jorge Valdivia Chavez

**Abstract**—The following article has the goal to present a practical application of computer vision in a real life problem. In the context of 2020 coronavirus pandemic, virtual education takeoff as the main option to enable the scholar systems to continue. Though, there exists various problems in the implementation of this technologies. For instance, this technologies don't have any proper tool to measure the level of attention of an audience. Here, we present an algorithm to measure the level of attention of an audience based on the emotions expressed by their facial expressions.

**Index Terms**—COMPUTER VISION, EMOTION CLASSIFICATION, EIGENFACES, FISHERFACES, LBPH

### I. INTRODUCCIÓN

LA incorporación de las tecnologías de la información a nuestras actividades cotidianas a sido una constante a lo largo del último siglo. Desde la concepción de la computación en el siglo XX, ésta se ha desarrollado a pasos agigantados y a logrado introducirse en toda industria que impera en el siglo XX. La necesidad de las tecnologías de información en algunos sectores ha tenido un acoplamiento menos natural, tal es el caso de la educación. Hasta hace un par de años, la adopción de las tecnologías de la información en el ámbito educativo se ha dado con una velocidad menor, la mayor parte de la educación se seguía impartiendo en las aulas, con la infalible tiza y un pizarrón. Es hoy, en el contexto de la pandemia de COVID-19, que para mantener, en medida de lo posible, a los profesores de todo el mundo impartieron clases se ha recurrido completamente a las T.I. Sin embargo, lo rápido que tuvo que darse el cambio, no permitió que estas tecnologías tuvieran todas las capacidades necesarias para generar la misma experiencia de enseñanza para los alumnos.

En el proceso de enseñanza que se vive en las aulas virtuales principalmente carece de la retroalimentación natural que se vive en un salón de clases presenciales. En un salón de clases presencial solo hace falta que el proceso vea un par de caras confundidas, o bien, note la distracción de sus alumnos en el lenguaje corporal que emite su postura en el pupitre para modificar la exposición de la clase. Pedir la participación espontánea de los alumnos, elevar el tono de la voz, algún gesto corporal puede ser determinante para captar la atención del público y hacer que la exposición de un tema se convierta en una mecánica que genere un verdadero proceso de aprendizaje.

Algunas aplicaciones que encabezan su uso en las aulas virtuales son zoom, teams, google meetings y discord. Todas

estas aplicaciones comparten funcionalidades básicas para la enseñanza en línea: transmisión de audio y vídeo, proyección de pantallas de alguno de los usuarios y un mecanismo para compartir una reunión con varios usuarios son algunas de las funcionalidades básicas que comparten estas aplicaciones. Sin embargo, la retroalimentación que se recibe sobre la atención del público está particularmente limitada a lo que se puede visualizar en las pantallas de las asistentes a la reunión que tengan su cámara encendida. Además que, al momento de compartir pantalla se minimiza la información que puede verse en las cámaras de los demás participantes a la reunión.

Dado que los videos de los demás participantes es el principal insumo para definir el nivel de atención de los participantes de la reunión es que se vislumbró un área de oportunidad para aplicar la Visión computacional como una herramienta para resolver el problema.

La visión computacional ha sido un área de conocimiento entre las ciencias de la computación que ha recopilado varias técnicas y mecanismos para generar algoritmos que repliquen a la visión natural en sistemas computacionales. La propuesta de solución es utilizar técnicas de visión para determinar emociones de los espectadores de la videoconferencia basadas en los fotogramas de los videos de las cámaras.

Para realizar esta tarea nos vamos a enfocar en métodos de reconocimiento facial basados en correlación: Eigenfaces, FisherFaces y LBPH. Los algoritmos se introducirán a continuación.

#### A. EigenFaces

La técnica de Eigenfaces nace como una alternativa para resolver el problema de almacenamiento de imágenes de rostros a través de generar representaciones de baja-dimensionalidad. La técnica se basa en el análisis de componentes principales, cuya idea fue introducida por primera vez por Sirovich Kirby (1987), utilizando una colección de imágenes para formar una base de características. A esta base de imágenes se les refiere como Eigenpictures y pueden ser combinadas para reconstruir imágenes del conjunto inicial.

Posteriormente, Turk Pentland (1991), propusieron utilizar los resultados de Eigenfaces para resolver el problema de

reconocimiento facial. Además de introducir un método para reconocer rostros de manera automática, mostraron un método para construir los eigenvectores y la matriz de covarianzas de tal forma que hace que el proceso de eigen descomposición fue eficiente para llevarse a cabo en un conjunto grande de imágenes.

La implementación de la técnica de Eigenfaces se pueden describir con los siguientes pasos:

- Preparar el conjunto de imágenes: Las imágenes a utilizarse con el Eigenfaces deben de haberse capturado con la misma iluminación y deben de normalizarse para tener las manos y la boca alineados en todas las imágenes. También es importante que las imágenes tengan una resolución en común. Las imágenes serán consideradas como un vector columna. Todos los vectores columna serán combinados en una matriz T donde cada columna es una imagen.
- Se resta la imagen media de cada vector columna de la matriz T.
- Se calcularan los eigenvectores y eigenvalores de la matriz de covarianzas S. Cada Eigenvector tiene la misma dimensionalidad que los vectores columnas y por tanto pueden ser vistos como una imagen. Por tanto, los eigenvectores de la matriz de covarianzas son llamados eigenfaces. Estos vectores son las direcciones en las que las imágenes se diferencian de la imágenes media. En general este paso es excesivamente costoso computacionalmente, por lo que la aplicación practica de Eigenfaces está determinada en encontrar un método eficiente para calcular los eigenvectores.
- Escoger los componentes principales. se ordenaran los eigenvalores en orden descendente y el número de componentes principales se determinará de tal manera que que k satisfaga  $(\lambda_1 + \lambda_2 + \dots + \lambda_k) / (\lambda_1 + \lambda_2 + \dots + \lambda_n) > \epsilon$

Notemos que los eigenfaces pueden ser utilizados ahora para representar las imágenes en base a estos eigenvectores.

### B. Local Binary Pattern Histogram (LBPH)

El método de patrones binarios locales fue diseñado para la descripción de texturas. Este algoritmo etiqueta los pixeles tomando en cuenta la distribución de sus vecinos.

En el caso de reconocimiento facial, para formar la descripción global del rostro, el algoritmo realiza descripciones locales, segmentando regiones del rostro y aplicándoles un histograma con el que se obtiene el operador LBPH que va a describir la información independiente por cada región. Con base en esto, se concatenan las descripciones locales para poder tener una descripción global.

Parámetros que utiliza:

- Radio. Se utiliza para construir los patrones circulares alrededor del pixel central (normalmente se le asigna el valor de 1).
- Vecinos. Número de puntos en la muestra para crear el patrón circular (normalmente se le asigna el valor de 8).
- Eje x. Número de celdas en dirección horizontal (normalmente se le asigna el valor de 8).
- Eje y. Número de celdas en dirección vertical (normalmente se le asigna el valor de 8).

El algoritmo que sigue este modelo es [2]:

- Utilizar un filtro (normalmente de 8x8) que recorra la imagen de manera iterativa seleccionando cada vez un pixel central y sus vecinos.
- El pixel se compara con cada uno de sus vecinos. Asignando 1 cada vez que el pixel central sea menor que el pixel comparado y un 0 en el caso contrario.

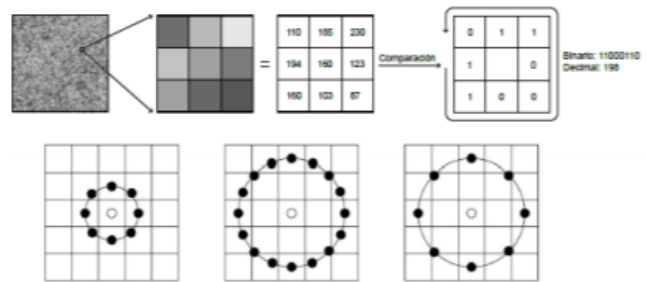


Fig. 1. Obtención de los parámetros LBPH.

- El número binario resultante se convierte en un número decimal que es contado en el histograma para formar la descripción.
- El histograma de las etiquetas de todos los pixeles es posteriormente utilizado como una descripción de la textura de la imagen.

### C. FisherFaces

La técnica de FisherFaces es una alternativa de la solución del problema de reconocimiento facial. La técnica puede ser descrita como una mejora sobre la técnica de Eigenfaces. Esta técnica se basa principalmente en el análisis de componentes principales y Análisis Discriminante lineal.

En párrafos anteriores, describimos el análisis de rostros a través de la técnica de eigenfaces. Notemos que en una imagen la iluminación juega como una característica que, para el análisis por eigenfaces, no fue caracterizado de manera distinta a demás componentes del rostro. Este problema del enfoque por análisis de eigenfaces es un

área de oportunidad que toma el análisis de FisherFaces en consideración en su construcción. FisherFaces extrae los componentes principales que diferencia a un individuo de otro.

El reconocimiento de imágenes que hace uso de esta técnica primero aplica el análisis de componentes principales para reducir la dimensionalidad de las imágenes y posteriormente aplica Análisis Discriminante Lineal. El Análisis Discriminante Lineal es usado para encontrar una combinación lineal de características que sea capaz de separar dos o más clases de objetos.

El algoritmo de ajuste de FisherFaces es el siguiente:

- Se debe de obtener imágenes de rostros.
- Se deberán de preprocesar las imágenes se deberán de pasar las imágenes a escala de grises y además se dividirá el conjunto de imágenes en un conjunto de entrenamiento y de prueba.
- Se aplicará un método para generar un vector de características de las imágenes de los rostros de conjunto de entrenamiento y los relacionaran con los elementos del conjunto de prueba haciendo uso de distancias euclidianas.
- Finalmente se extraeran las características de los rostros.

## II. DESARROLLO

A continuación se mencionan los pasos que seguimos para la implementación de una solución que busca detectar emociones en los estudiantes:

- Para la detección del rostro, utilizamos la función "Cascade Classifier Training" de Python. Esta función utiliza las características Haar-like, que es un procedimiento de detección de objetos que clasifica las imágenes basándose en el valor de las características simples, después de esto, utiliza una cascada "Attentional", que es Es un árbol de decisión degenerado donde en cada fase un detector es entrenado para detectar casi todos los objetos de interés mientras rechaza a los que no lo son.
- Creación de dataset personalizado, que segmento por emociones que incluyen: felicidad, sorpresa, enojo y tristeza Fig. 2.
- Preparación de entorno de entorno de trabajo, utilizando herramientas como Python, Open CV.
- Se programó el modelo de EigenFaces para hacer la clasificación
- Se entrenó el modelo de EigenFaces con el dataset obtenido
- Se incorporaron los modelos de FisherFaces y Local Binary Pattern Histogram (LBPH) desde Open CV
- Se entrenaron los dos algoritmos restantes
- Se crea el streaming con los 3 modelos para evaluar su desempeño con una persona frente a la cámara

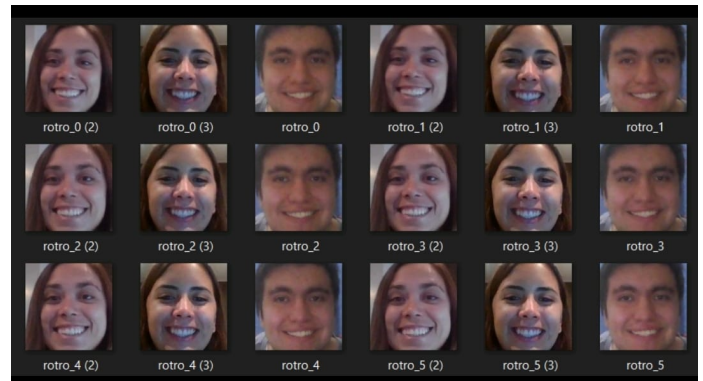


Fig. 2. Ejemplo del dataset Entrenado.

## III. RESULTADOS

El resultado de esta implementación es un algoritmo que permite clasificar las emociones en tiempo real. Gracias a las herramientas podemos tener un video con la retroalimentación de la emoción como se aprecia en las imágenes 4 y 3

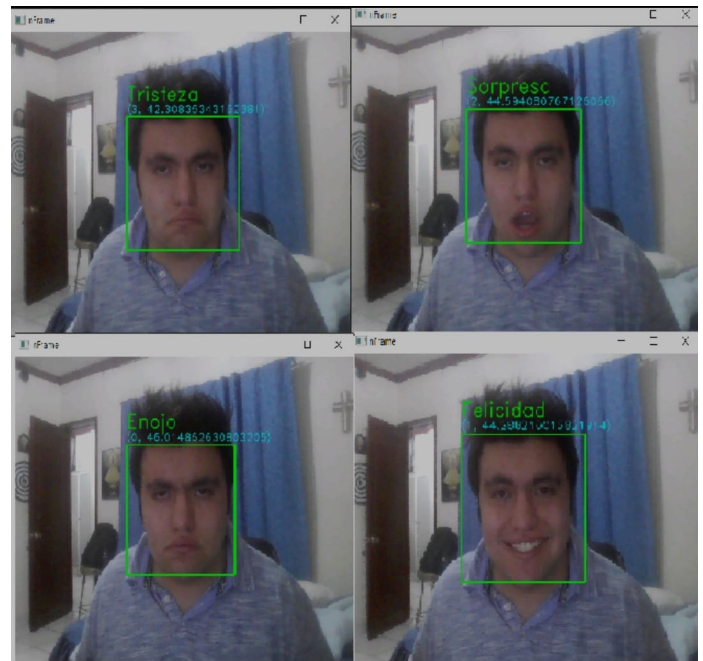


Fig. 3. Imagen panorámica resultante.

## IV. CONCLUSIONES

Con la elaboración de este proyecto, logramos comprobar, entre otras cosas, las principales diferencias entre los métodos de Eigenface, Fisherface y LBPH.

Pudimos comprobar que los dos modelos que utilizan correlación para su aplicación son Eigenface y Fisherface, pues toman un enfoque holístico para la identificación facial.

El método de Eigenface toma como base un análisis de componentes principales para disminuir la dimensionalidad

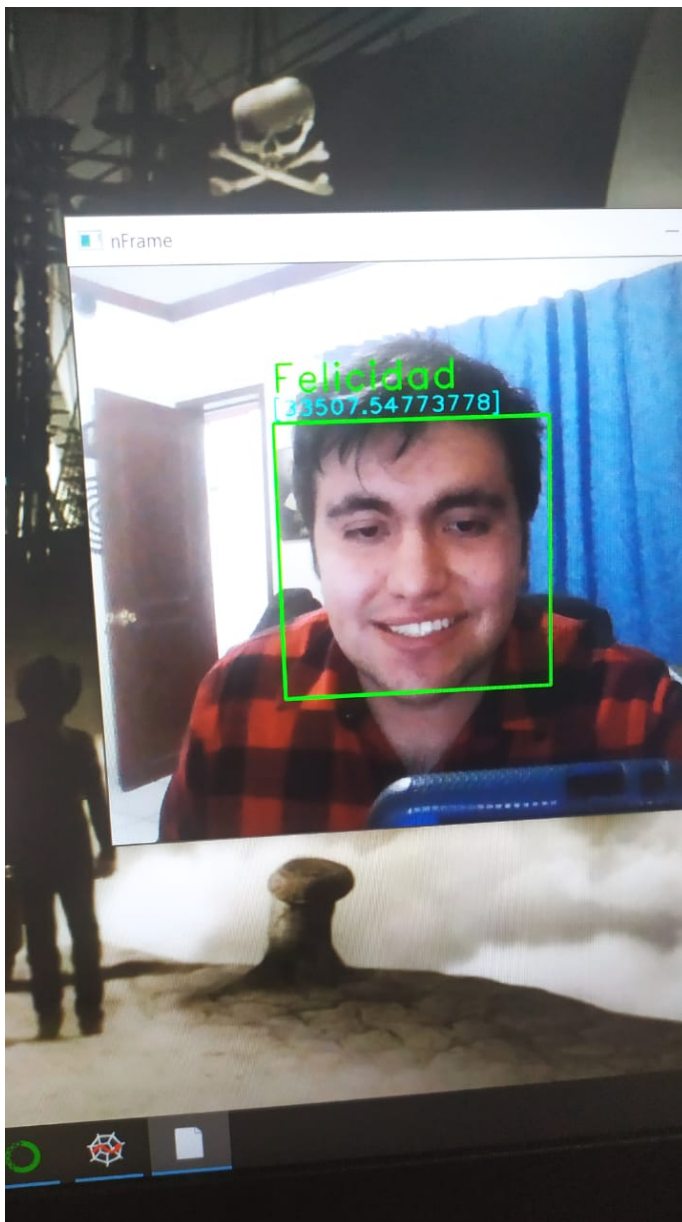


Fig. 4. Imagen panorámica resultante a color.

de las características; mientras que el método de Fisherface utiliza el método del Discriminante Lineal para disminuir la dimensionalidad.

El método de Eigenface puede llevar problemas cuando la varianza se deba a componentes externos, pues los componentes con una varianza máxima sobre todas las clases existentes podrían no ser necesariamente útiles para la clasificación. El método Fisherface, toma esto en cuenta, y, para preservar información discriminativa aplica la Discriminación Lineal.

Por otro lado, el método de LBPH, no utiliza esta misma visión. En este caso, el algoritmo busca describir únicamente las variables locales de un objeto, comparando el píxel central con los píxeles vecinos.

Finalmente, una de nuestros principales obstáculos en el entrenamiento de las imágenes fue al momento de querer diferenciar las caras entre masculino y femenino, pues finalizaba con mayores inconsistencias a cuando únicamente entrenábamos un género.

## V. CÓDIGO

Se puede consultar el código en el siguiente link:  
[https://github.com/magojval/Vision\\_Proyecto](https://github.com/magojval/Vision_Proyecto)

La base de datos y la demo se pueden consultar en el siguiente link:

<https://drive.google.com/drive/folders/1TujBewPCCGxsKbWI4FRKnS>

## REFERENCES

- [1] LBPH. Silva, Esparza, Mejía, 2012)
- [2] Esparza, C., Tarazona, C., (2017) Reconocimiento Facial Basado en Eigenfaces, LBPH y Fisherfaces en le Beagleboard - xM, Revista colombiana de tecnologías de avanzada (RCTA)
- [3] Ottado, G. Reconocimiento de caras: Eigenfaces y Fisherfaces
- [4] Sierra, M., (2015), Estudio comparativo de modelos de identificación facial basados en correlación