

Projeto Ticket Face: Bilheteria por Reconhecimento Facial

Mitson D. G. Ferreira,¹ Lucas V. F. da Silva,² Jocivania Pinheiro³

Centro de Ciências Exatas e Naturais, Ufersa, Mossoró, RN

Com a chegada da pandemia da COVID19 os setores de eventos foram os mais impactados justamente por lidarem com uma grande quantidade de públicos aglomerados em filas e no local dos eventos. O “Ticket Face” vem justamente para resolver o problema de aglomeração em filas, dando mais rapidez, melhorando a experiência do usuário, resguardando a saúde do mesmo e também dos funcionários da empresa, pois, não será necessário o usuário ter contato com pessoas para adquirir o ingresso ou até mesmo entrar no dia do evento. Uma vez que teremos uma plataforma totalmente online para compra dos ingressos e cabines automáticas de reconhecimento facial na entrada do evento. Vale ressaltar, que a tecnologia de reconhecimento facial tem se tornado cada vez mais comum em todo o mundo [3].

O Projeto que denominamos de “Ticket Face” é mais do que um simples projeto, é um modelo de negócio que serve não apenas para ajudar a população com os problemas de filas para a entrada de espaços públicos como também é um produto empresarial para comercialização “B2B” (Business to Business) através dos nossos canais de vendas. (Conheça mais clicando no link: <http://www.bit.ly/ticketface>) Contaremos com a plataforma “Ticket Face” totalmente online, veja Figura 1, onde o usuário irá fazer seu cadastro, selecionar seu ingresso e efetuar o pagamento. Em seguida será direcionado para um campo onde enviará um arquivo com a foto do seu rosto, esse arquivo entrará em nosso banco de dados e em seguida servirá de parâmetro para comparação dos rostos e autenticação do portador do ingresso, para que o mesmo possa acessar no dia do evento.



Figura 1: Imagem do Site.

O reconhecimento facial tem como base pegar uma foto de uma pessoa e extrair dados dessa foto. Desta forma, é bem comum o uso de matrizes, à álgebra linear, mas especificamente que consiste em reorganizar esses dados extraídos da foto em um vetor e assim colocando outros rostos temos uma matriz com esses vetores. Para tal, usamos um sistema embarcado chamado ESP32Cam que já provém um sólido e produtivo ambiente com câmera e um software programável que nos permite adaptar a nossas demandas. Dito isso, a parte da matemática será explicada nos próximos parágrafos.

Precipuaente, o software decompõe a imagem (rosto de uma pessoa) nas suas componentes vermelho, verde e azul, onde cada uma das cores é uma matriz 3×3 . Em seguida ele cria uma imagem

¹mitson.ferreira@alunos.ufersa.edu.br

²lucas.silva69305@alunos.ufersa.edu.br

³vaniamat@ufersa.edu.br

média das apresentadas, organizando os espaços vetoriais em olhos, nariz, boca, entre outros. Isso ocorre com aplicação da análise dos componentes principais (ACP)[1], visto que uma imagem vai gerar um vetor muito grande, esse método diminui esse espaço em componentes principais, a exemplo, como já mencionados boca, nariz, entre outros. E, para uma melhor porcentagem de acertos, fazemos com que cada imagem contribua igualmente para o sistema, seguindo esse cálculo:

$$Valor_{novo} = \frac{Valor_{antigo} - Média}{Desvio\ padrão}, \quad (1)$$

onde a *Média* é dada pelo quociente entre a soma de todos os rostos na sua forma vetorial e a quantidade total de rostos.

Agora, fazemos o cálculo de matriz de covariância, pois ela busca estabelecer alguma relação ou ausência de relação entre as variáveis de um determinado dataset (seria um banco de imagens). Com os dados já colocados em matriz de covariância, agora calculamos os autovetores e autovalores para identificar os componentes principais. Os principais componentes são novas variáveis que foram construídas pela combinação linear dos dados iniciais. Por fim, basta multiplicar a transposta da matriz com os autovetores da matriz de covariância pelos dados já associados uns aos outros.

$$dataset_{final} = (autovetores_{cov})^T \cdot dataset_{padronizado}. \quad (2)$$

Com isso nós temos os dados em uma dimensão com uma porcentagem de informação, e por fim aplicamos o método Eigenfaces [2], que basicamente cria uma imagem média através dos componentes principais e assim podemos comparar essa imagem média pelas demais, através da seguinte fórmula $F = F_m + \sum_{i=1}^n \alpha_i F_i$, onde F é uma nova face, F_m é a face média, F_i é uma eigenface e α_i é uma escalar para encontrar novas faces, podendo ser negativas ou positivas.

O trabalho está em desenvolvimento usando as ferramentas Nodejs, que é uma plataforma responsável por executar javascript para a criação da lógica de todo o sistema. Já para o reconhecimento facial estamos usando a biblioteca OpenCV, é com ela que podemos executar todo o processo acima, nos quais, utilizaremos amostras (conjunto de imagens para ser processados pela OpenCV) obtidas da internet. Por fim, estamos trabalhando para resolver o problema das inconsistências ao considerar todas as etnias.

O Ticket face foi necessário, primeiramente, para provar que com estudos e dedicação é possível conseguir criar coisas grandiosas. O próximo passo será amadurecer o nosso projeto para que vire realmente um produto comercial B2B (Business to business) e futuramente nosso principal objetivo é aprimorar nossa tecnologia de reconhecimento facial para que seja usada em demais áreas e colaborando positivamente para a sociedade em diversos setores.

Agradecimentos

Os autores agradecem o apoio da UFERSA na execução deste trabalho.

Referências

- [1] MEDIUM. **Reconhecimento de padrões - Eigenfaces**. Online. Acessado em 15/10/2021, <https://medium.com/@williangp/reconhecimento-de-padrC3B5eseigenfaces-e4cef8f04919>.
- [2] SINFIC. **Reconhecimento de padrões através de Eigenfaces**. Online. Acessado em 15/10/2021, <http://www.sinfic.pt/SinficWeb/displayconteudo.do?numero=44666#Conteudo>.
- [3] SURFSHARK. **The Facial Recognition World Map: Smile you're on câmera**. Online. Acessado em 14/08/2021, <https://surfshark.com/facial-recognition-map>.