Edge AI Face Recognition for Public Transport Fare Payment

SUBMISSION DATE / POSTED DATE

29-10-2022 / 02-11-2022

DOI

10.36227/techrxiv.21432000.v1

Edge AI Face Recognition for Public Transport Fare Payment

Blaise Marvin Rusoke
Dept. Electrical and Computer Engineering
Makerere University
Kampala, Uganda
blaise.rusoke@students.mak.ac.ug

Denis Musinguzi
Dept. Electrical and Computer Engineering
Makerere University
Kampala, Uganda
denis.musimguzi24@students.mak.ac.ug

Simon Peter Miyingo
Kiira Motors Corporation
Kampala, Uganda
simon.miyingo@kiiramotors.com

Andrew Katumba
Dept. Electrical and Computer Engineering
Makerere University
Kampala, Uganda
andrew.katumba@mak.ac.ug

***Abstract*—Face Recognition technology is chiefly concerned with accurately re-identifying individuals through the use of mathematical face representations. It presents a window of opportunity for the introduction of a fast, automated, seamless and easy to deploy form of biometric technology. In this research we design a fast, easy to use, and privacy oriented contactless payment system for public transportation that chiefly makes use of face recognition and internet of things technologies. We demonstrate a one-shot face recognition model and also prepare and test it for real-time inference on the edge.**

Our system makes use of a Siamese Model built on top of the Inception-Resnet V1 architecture with accuracy, precision and recall values of 93.81%, 90.91% and 97.35% on our validation set.
The model was deployed on a Raspberry Pi 4 Model B with an Intel Neural Compute Stick 2. Inference was performed through the inference engine API of the OpenVINO toolkit on the Neural Compute Stick plugged into the Raspberry Pi.
The system is composed of three other subsystems, i.e. the edge device, cloud database and user interface subsystems which work together to ensure that payment is complete in under 2 seconds.

***Index Terms*—Siamese Network, Edge device, Face Recognition, Payment system**

I. Introduction

There's no doubt that public transportation is a crucial factor in the urban, suburban and rural ways of life. For the vast majority, it's not just a convenience but a necessity. A necessity important enough to bolster the economy through giving agency to people to move for business and pleasure.

The sector certainly has its ups and downs; terrible road networks, high fuel prices, congestion and long commute times have all taken their toll. Furthermore, the challenge of having an efficient, safe, and convenient form of public transport has for long been a hard task to tackle. Inconvenient payment techniques and architectures are still prevalent in the sector. This has left many customers frustrated and unsatisfied after having to carry paper tickets and make long queues to have them verified. Countless bus owners and stakeholders also continuously fail to get their investment's worth as a result of employing middlemen to bridge the cash flow gap between them and the customers. The recurring COVID-19 waves massively exposed the sector and in-turn led to its closure not only in Uganda but in many other parts of the world.

There's now clearly need for reinvention in this sector before the proverbial train sets off. [17, 27]

The case study for this work is the Kayoola EVS bus, a fully electric low-floor bus for urban mass transportation manufactured by Kiira Motors Corporation, Uganda. Kiira Motors Corporation is a state owned enterprise incorporated to develop, make and sell vehicles, ultimately focusing on Automated, Connected and Electric (ACE) vehicles [5, 6].

Central to this research is the desire to make use of A.I based face recognition technology to spearhead the public transport sector reinvention in Uganda.

Currently, top performing face recognition based solutions such as Apple's FaceID make use of 3D face representations and are invariant to lighting. As explained in section II majority of the cited face recognition solutions approach the face recognition challenge as a multi-class classification task and make use of traditional machine learning algorithms like Support Vector Machines.
In regards to contactless payments in Uganda, RFID and NFC are the commonest technologies [9].

This research makes use of 2D face representations and avoids the overhead costs associated with using 3D face representations, i.e. the pertinent sensors and computing power. We also use a Siamese network, a more pragmatic approach to real world face recognition challenges than the multi-class classification approach used in majority of the cited solutions in section II. With the Siamese network, the model needs not to be retrained each time a new user enrolls

into the system and better still there's no need to request for more than one image from the user. This reinforces our system target of promoting user data privacy.
Lastly, we successfully deployed the core part of the system at the edge on a Raspberry Pi with inference powered by the Intel Neural Compute Stick 2. This was an inexpensive and time sensitive approach that was also easy to use and install. This edge solution also eliminated the non-pragmatic approach inhibited by latency posed by majority of the cited face recognition solutions in section II.

Compared to NFC and RFID technology, our approach maximizes user convenience by eliminating the need to constantly carry payment cards or tags. Your face becomes your ID.
It's also cheaper to install, and eliminates the security risk posed by RFID and NFC cards/tags because of their inherent ability to be reprogrammed [34, 19] .

II. Literature Review

Contactless payment methods are fast evolving with the most popular approaches making use of debit or credit cards, smart cards, NFC and RFID technologies. However, not all available contactless payment technologies are suited for public transportation in the Ugandan setting.
In this section we discuss popular contactless payment systems and related face recognition solutions. We highlight the gaps in this work which we then subsequently solve through our research's methodology.

A. Contactless Payments using NFC and RFID

Radio Frequency Identification (RFID), is a technology for automated identification of objects and people. Items are marked with RFID tags which contain transponders that emit messages readable by specialized RFID readers [33, 20]. To make payments, contactless credit cards use RFID technology when they communicate with readers at radio frequencies [23]. RFID contactless cards are widely used to facilitate contactless payments by top brands like Mastercard, VISA, and Oyster [23, 33].
Smart contactless RFID cards are only however able to work in a 10cm range. Worse still if a card falls into the wrong hands, its easy for someone else to make a lot of transactions as they don't need to know the owner PIN, nor is there a limit on the number of transactions that can be made. The associated high installation costs and high risk of electronic pickpocketing all discredit RFID technology [2, 20, 8]

On the other hand, Near Field Communication, NFC is a short range wireless communication technology that facilitates communication between NFC-compatible electronic gadgets [19]. To communicate, NFC-enabled devices can be placed in contact with or in the vicinity (maximum separation of 4cm) of other NFC-enabled devices. NFC technology is superior to RFID and has found application in the public transportation sector where its used to facilitate fast digital ticket validation.

Research by Nasution et al. [26] illustrates the development of a Train NFC application for railroad public transportation to facilitate the process of purchasing train tickets and electronic ticket distribution and validation through cell phones.

Ghiron et al. [10] and Tamraker et al. [31] also illustrated the use of NFC in public transportation systems through digital tickets that could be validated on cell phones using NFC.

The idea of the use of digital tickets and NFC technologies for airplane tickets has also been proposed and evaluated at the Yogyakarta International Airport [11].

NFC systems are prone to security attacks during transmission which can be categorized as Eavesdropping, Data corruption, Data modification, Imposter attacks and Theft i.e. where an NFC device gets stolen. [19]. For eavesdropping, an attacker uses an antenna to record communication between NFC devices. The intention of such an attack is usually to corrupt or modify the information being exchanged.
Imposter (Man in the middle) attacks are also possible regardless of NFC's proximity standards. Attackers still can intercept information, manipulate it and relay it to the receiving device. [32, 25, 4] NFC infrastructure is also very expensive to install, and the technology isn't supported by all smartphones.

B. Face Recognition Based Systems

The ubiquity of camera devices, and advancements in Deep Learning have all contributed to the widespread adoption of face recognition technology.
Today, we have automated retail stores, advanced biometric systems, sophisticated recommendation systems, all as a result of advancements in face recognition technology.

Maheen Zulfiqer et al. [35] presented a deep learning based face recognition biometric authentication system. Their research made use of the Viola Jones algorithm for face detection and tried out a number of CNN architectures before settling with the SqueezeNet [16] algorithm due to computational costs and accuracy (98.76%).
In their implementation technique, the pre-trained SqueezeNet model was used together with a deep neural network with a softmax output. The process involved using the augmented multiple images collected for each identity to fine-tune the SqueezeNet model.

Because the system used the multi-class classification approach, multiple images are required from each user both during the development and production phase. This violates user privacy and also increases storage costs during production. Because of the implementation technique, the system has to be retrained each time a new user is added. This clearly isn't pragmatic for real world use cases.

Chao and Yang et al. [12] combined face detection and recognition algorithms to build a video based face recognition system for efficient and accurate recognition of faces of specified characters in videos. The system made use of the MTCNN face detection model to detect and extract faces and the FaceNet [29] face recognition model to extract pertinent face features. A Support Vector Machine was thereafter utilized to identify the different individuals in the videos and the entire process embodied a multi-class classification task. Because of the recognition task's specificity, a custom face dataset was created by crawling target individuals' images off the internet. The accuracy of this video based recognition system on the custom dataset was 94.9%.

This implementation focuses on training an SVM classifier with a pre-trained FaceNet model. The system henceforth is at risk of reinforcing A.I gender and racial bias since the FaceNet model isn't retrained nor fine tuned. The multi-class classification approach also implies that the system would have to be retrained each time a new user is to be added. Multiple image examples would have to be collected for each user, even in the production phase. This puts the system at risk of violating user privacy and makes storage and computation expensive.

Ferry Cahyono et al. [3] discussed the appropriate method to be applied in an employee presence system using faces by comparing two deep learning architectural models, FaceNet [29] and OpenFace [1]. FaceNet and Openface models are used to extract 128 dimensional facial features from the employee image dataset. The classification of the facial features is then performed using the Support Vector Machine (SVM) model. To validate this model, 5-fold cross validations is used. From this research, it was observed that FaceNet accuracy results were higher with a perfect accuracy of 100% than Openface results with a 93.33% accuracy.

This implementation focused on selecting the right face recognition model. The use of an SVM classifier however thwarted all pragmatic deployment plans of this work. The system would have to be retrained each time a new user is to be added, and multiple image examples collected.

Saibal Manna et al. [24] proposed a face recognition system that makes searching for criminals easy and quick with less time. This research made use of a pre-trained FaceNet [29] model to transform input images into a close-packed Euclidean space. The execution was centered on three significant steps, the first step was to find a decent database of human faces with multiple images per identity, the second stage was to identify and prepare the faces in the dataset using OpenCV and the MTCNN face detection model and lastly the faces were mapped to a 128 dimensional Euclidean space using the FaceNet [29] model. In order to correlate face similarity and in turn recognize the different individuals, square L2 distances of the generated embeddings were computed in the Euclidean space. Images belonging to the same individual had smaller distances between their embeddings and those belonging to different individuals had much larger distances between their embeddings.

The lack of training or fine-tuning of the FaceNet model in this proposal however puts this system at risk of reinforcing racial and gender bias in A.I models.

Byung-Gil Han et al [14] presented a passenger management system based on face recognition technology for intelligent transport vehicles. The management system made use of the AdaBoost method utilizing Local Binary Pattern histogram for face detection and the face recognition task was designated to a remote server. In this research most of the emphasis was placed on the development of a fast face detection system with a detection time of 44.6ms.

For this task, emphasis is placed on developing a fast face detection algorithm. The absence of face recognition model training implies that the system's results could be prone to prejudices when exposed to diverse unseen data. The use of a remote server for face recognition implies that the system will have a high latency during operation and is therefore not suitable for real time operation.

Sarathi et al [28] discussed a fare management system for transport corporation using face recognition based on principal component analysis. The system used the viola jones algorithm for face detection. Face recognition was performed by a remote server using the Principal Component Analysis method. It's key to note that this implementation stored a stack of images on their remote database which were used to train, test and validate the system implemented in Matlab.

The system made use of a remote server for face recognition, it therefore is prone to having a high latency during operation and not pragmatic for real-time tasks. The storing of multiple user images on the system's cloud database may lead to high storage costs during production and put the system at risk of violating user privacy.

Our solution performs inference on the edge using a Raspberry Pi microcontroller and the Intel Neural Compute Stick. It is henceforth not prone to latency issues during real-time operation. The system also makes use of a Siamese model, hence overcoming the challenge of having to collect and store multiple user images even during production, and the need to be retrained each time a new user is enrolled to the system.
Uncommon to all other cited solutions, our system also uses an anti-spoof detection model[7] which enhances the overall system's security by blocking spoof attacks.
Lastly, only a single face image is required for each user, and this is encoded into a string representation so as to enhance user privacy and lower storage costs.

III. Methodology

This section details the steps taken in building the A.I based face recognition system. It discusses the data collection and preprocessing phases, the model architecture design and training process, and finally edge deployment and the pertinent general system setup.

A. Data Collection

To build our datasets, our initial approach was to make use of the willing individuals in our local society. To successfully harvest this data, the entire collection process was automated using an in-house written python script. The script made use of python's OpenCV library, a pre-trained face detection model, and the python os module. With the pre-trained model and the OpenCV library, a face was detected, localized and extracted from the live video camera stream. The webcam of a LENOVO i3-7th Generation 4GB RAM laptop was used. The python os module was then used to store these images to disc, with each directory corresponding to a distinct user. The collected data is described below

Table I: Locally Collected Data

Females	3
Males	9
Image Composition	5,808 images
Average per Identity	484 images

Because of the quality of the data and ease of the collection process, we also manually scrapped data from the internet to create more datasets. Three datasets, which predominantly contained black images, were created over the course of the project.
The datasets are described below.

Table II: Data Scrapped off the Internet

Composition	Dataset1	Dataset2	Dataset3
Distinct female Identities	6	44	35
Distinct male Identities	9	29	24
Images Composition	1,474	1,346	3,270

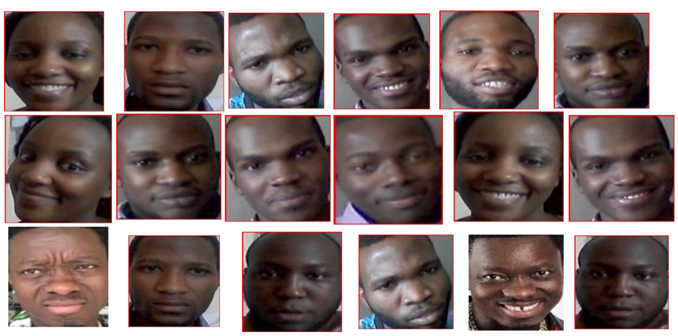

Figure 1: Some of the images contained in our datasets

B. Data Preprocessing

Data preprocessing on the collected data was performed sequentially through the steps indicated below

1) Detection, Localization and extraction of faces from images using a pretrained face detection model.
2) Rescaling of face data (divide by 255)
3) C: Data Standardization (subtract the mean, and divide by the standard deviation)
4) Convert color format to RGB
5) Store processed data to disc

C. Model Building and Training

This section details the model building and training process of the Siamese and MobileFaceNet-arcface models which were considered for this task.

1) Siamese Model

A Siamese Model[22] was built using an Inception-Resnet V1[1] implementation of the FaceNet model[29]
As illustrated in figure 2,the FaceNet model was used to implement the identical subnetworks of the siamese model.

Since the two FaceNet based sub networks output 128 dimensional embeddings. The Euclidean distance between the output embeddings of the two identical sub networks was computed and the output was then fed into a Fully Connected layer with a single node and a sigmoid activation function. This ensured that values were close to 1 for a perfect match, and 0 for mismatch.

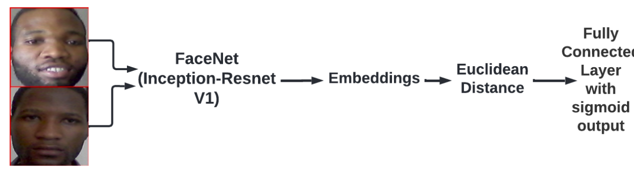

Figure 2: The Siamese Model Architecture

Training
The images in the training and testing data clusters were rearranged into positive and negative image pairs. Positive image pairs were comprised of images belonging to the same class (identity), whereas negative image pairs were comprised of images belonging to different classes (identities). Positive image pairs in both the training and test data clusters were assigned a label of 1. Negative image pairs were assigned a label of 0.

Figure 3: Positive Image Pairs

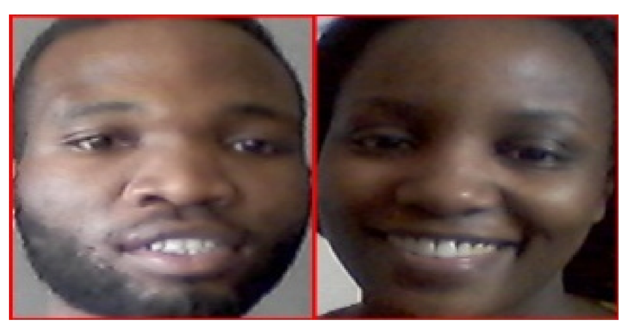

Figure 4: Negative Image Pair

[1]https://github.com/nyoki-mtl/keras-FaceNet

The model was trained on 737 image pairs belonging to 15 identities, from dataset 1 described in table II. The data fed to the model was randomly sampled from the positive and negative image pairs contained in the training and test data clusters. It is through these pairs that the Siamese model learned to quantify image similarity using the Binary Cross Entropy loss. Table III summarizes the set parameters during training

Table III: Key hyper-parameters

Train-test split ratio	12:1
Learning rate	0.001
Optimization algorithm	Adam
Activation function	ReLU
Epochs	350
Base CNN architecture	Inception-Resnet V1

Throughout the training process, accuracy, loss, precision, recall and AUC (Area under the Curve) were the metrics tracked.

Table IV: Siamese Model Training Results

Mode	Accuracy	Loss	Precision	Recall	AUC
Training	91.48%	21.65%	91.91%	90.96%	97.36%
Validation	93.81%	18.80%	90.91%	97.35%	98.42%

2) MobileFaceNet

A pre-trained model, MobileFaceNet[18] was used in this implementation.

The goal for this task was to fine tune the pre-trained model by performing a multi-class classification face recognition task.

A set of fully connected layers, and an output softmax layer were appended to the model so as to process the pre-trained model's embedding output for the multi-class classification task.

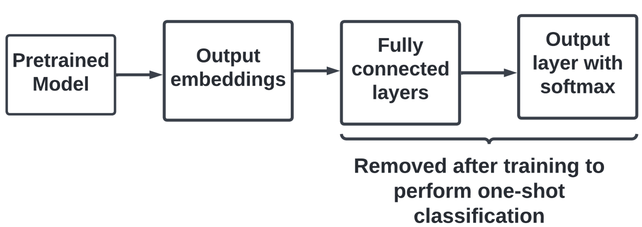

Figure 5: MobileFaceNet fine-tuning architecture

The architecture in figure 5 was developed such that the fully connected layers could be easily removed, and the model deployed as a one-shot type of model after training.

Training

The model was fine-tuned on dataset 3 described in table II Some of the key hyper-parameters are shown in table V

Metrics such as loss, accuracy and precision were tracked throughout training. The added fully connected layers after training were removed from the model which left the 128-dimensional vector embedding output as the final output layer of the model. This was done so that the model could be used for one-shot recognition tasks.

Table V: Key hyper-parameters

Train-test split ratio	12:1
Learning rate	0.001
Optimization algorithm	RMSProp
Activation function	ReLU
Epochs	200
Batch size	100
Loss function	Cross entropy

Table VI: MobileFaceNet-arcface training results

Accuracy	96.6%
Precision	100%
Recall	100%
Loss	2.8%

3) Siamese Model v.MobileFaceNet-arcface

Much as both models can be used in one-shot tasks, it was observed that the Siamese model was more robust than the MobileFaceNet-arcface model. For the same one-shot learning tasks, the MobileFaceNet-arcface model generated more false negatives than the Siamese model as depicted in table VII. This effect was investigated and attributed to variations in age of the identity, ridiculous makeup overhauls and also wild variations in face pose and orientation. This implies that in order for the MobileFaceNet arcface model to recognize an individual in a one-shot learning type of task, the current face, pose and orientation have to be greatly similar to those in the original anchor image. This for the most part renders the entire system impractical in the real world.

Therefore in conclusion, the Siamese Model was chosen over the MobileFaceNet-arcface model because of its higher Recall values. In terms of Precision, both models performed excellently.

Table VII: Siamese Model v. MobileFaceNet-arcface

Model	Precision	Recall
Siamese Model	90.91%	97.35%
MobileFaceNet-arcface	94%	71.85%

D. Edge Deployment

1) Model Optimization

Using the model optimizer command-line tool of the OpenVINO toolkit, the trained Siamese model was optimized for real-time inference through the tool's default hardware agnostic optimizations such as layer quantization, freezing, fusion and batch normalization. The model's weights were also converted from the 32 bit floating point numbers to 16 bit floating point numbers. This is because it's the 16 bit format that is supported on the Intel Neural Compute Stick.

2) *System Control Algorithm*

The operation of the system's control algorithm is as highlighted below

1) A pretrained face detection model extracts faces from a live video stream of boarding passengers
2) A pretrained anti-spoof[7] detection model determines whether or not the extracted face is real
3) Real faces are recognized, and the trip fare is deducted from the user's wallet balance
4) Spoof attack detected feedback issued on the Human to Machine Interface. Payment is also blocked.
5) New wallet balance is reflected on the user's mobile application and a positive response is also issued on the Human to Machine Interface.

3) *System Database Design*

From the system's control algorithm, there was need for a remote database system that stored the required user data and also was a base for updates to the user application and human to machine interface on the bus.
A non-relational database was designed using google firebase and it stored the users' wallet balance, face data, email, phone number and other pertinent meta data. User face data was stored in form of base64 encoded strings.

4) *Development of the Edge Device System*

The control logic, Siamese model, and the pre-trained face detection and anti-spoof models were deployed on a raspberry pi model 4B, 8GB RAM. A casing for the system was designed using Solid Works [30], a computer aided design software. The CAD model was thereafter 3D printed. Figure 6 shows the complete edge device system. The Raspberry Pi with the control software, the Intel Neural Compute Stick 2 to accelerate inference at the edge of the optimized model, all enclosed in the designed casing. During system tests after deployment, passenger video data was read into the system using a phone camera through the IP Webcam android app

Figure 6: The 3D Packaged Edge Device System: A Raspberry Pi microntroller and the Intel Neural Compute stick enclosed in a 3D printed casing.

A Human to Machine Interface, powered by the edge device was built to make it easy for users to interact with the deployed system.
Users simply had to face the camera, and then visualize the system's feedback on the HMI. The HMI issued feedback when payment was blocked due to a detected spoof attack, when payment was successful and complete and also when a user's wallet balance was insufficient for the trip.
The current trip's fare amount was also visible on the HMI. The HMI was developed in the python kivy framework[21], and also ran on the Raspberry Pi on the bus.

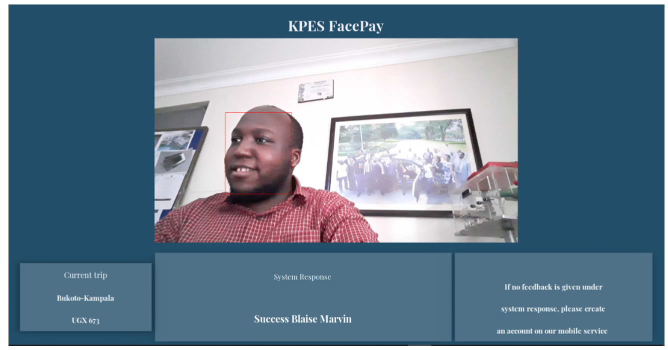

Figure 7: The Human to Machine Interface

5) *Development of the User Interface*

The system's user interface was developed using the kivy[21] framework in the python programming language. The user interface primarily facilitated user digital wallet top-up through credit and debit cards, and also user face data scanning. The interface made use of a face detection model that transformed a user's submitted face data into an encoded string which was later stored on the cloud database. Through the user interface a user was also able to visualize their wallet balance.

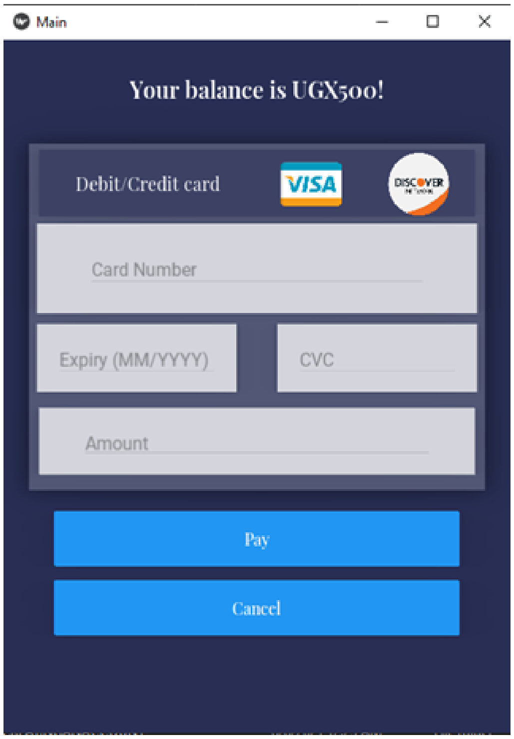

Figure 8: The payments page of the developed user interface

IV. Discussions and Limitations

The developed face recognition payment system took no more than 2 seconds to complete a payment, as observed during the deployment tests. This places the system on level terms with smart card technology with RFID/NFC.
However with NFC/RFID smart card payments, a transaction takes 2-4 days to be reflected on a user's account, compared to a maximum of 2 seconds in our payment system.

Majority of the studied face recognition systems e.g Zufliqer et al. [35], Yang et al. [12] and Cahyono et al [3] approached face recognition as a multi-class classification task. Accuracy was the metric tracked in these systems and the recorded values were 98.76%, 94.9% and 93.33% respectively.
Our system, employed a one-shot recognition technique, that tracked the accuracy, precision, and recall as shown in table IV
In comparison to these systems, and the 95.12% FaceNet accuracy on the YouTube faces dataset, our results lie within the acceptable ranges for face recognition.

Our system's pipeline makes use of a pre-trained anti spoof model based off research by Costa et al. [7]. During our deployment tests, this model efficiency deteriorated with variations in lighting, and with low image quality.

The siamese network used was trained using the binary cross entropy loss. The contrastive [13] and triplet loss [15] functions are however better suited for one-shot recognition tasks and produce better results. Furthermore, the use of 3D face representations has greatly improved the accuracy of face recognition systems as evidenced by Apple's FaceID.

Future attempts of this project will be address these highlighted problems.

V. Acknowledgment

The authors would like to thank the management of Kiira Motors Corporation for funding this project.

References

[1] Tadas Baltrušaitis, Peter Robinson, and Louis-Philippe Morency. "Openface: an open source facial behavior analysis toolkit". In: *2016 IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE. 2016, pp. 1–10.

[2] Erik-Oliver Blass et al. "PSP: private and secure payment with RFID". In: *Proceedings of the 8th ACM workshop on Privacy in the electronic society*. 2009, pp. 51–60.

[3] Ferry Cahyono, Wirawan Wirawan, and Reza Fuad Rachmadi. "Face recognition system using facenet algorithm for employee presence". In: *2020 4th International Conference on Vocational Education and Training (ICOVET)*. IEEE. 2020, pp. 57–62.

[4] Cheng Hao Chen, Iuon Chang Lin, and Chou Chen Yang. "NFC attacks analysis and survey". In: *2014 eighth international conference on innovative mobile and internet services in ubiquitous computing*. IEEE. 2014, pp. 458–462.

[5] Kiira Motors Corporation. *Kayoola EVS*. https://www.kiiramotors.com/kayoola-evs/. [Online; accessed 31-January-2021]. 2020.

[6] Kiira Motors Corporation. *Who We Are*. https://www.kiiramotors.com/about/. [Online; accessed 31-January-2021]. 2020.

[7] Valter Costa, Armando Sousa, and Ana Reis. "Image-based object spoofing detection". In: *International Workshop on Combinatorial Image Analysis*. Springer. 2018, pp. 189–201.

[8] Wei Dai et al. "Analyze on mobile payment based on RFID". In: *Procedia Environmental Sciences* 10 (2011), pp. 950–955.

[9] Kikonyogo Douglas Albert. *In just a single year of existence, Quick Tap is revolutionizing cashless payments in Uganda*. https://www.dignited.com/25564/quick-tap-cashless-payments/. [Online; accessed 31-January-2021]. 2017.

[10] Stefano Levialdi Ghiron et al. "NFC ticketing: A prototype and usability test of an NFC-based virtual ticketing application". In: *2009 First International Workshop on Near Field Communication*. IEEE. 2009, pp. 45–50.

[11] YVES I GONZALES. "Application of near field communication technology for mobile airline ticketing". In: *Journal of Computer Science* 8.8 (2012), pp. 1235–1243.

[12] Chao Guo and Yingyun Yang. "Implementation of a specified face recognition system based on video". In:

2019 IEEE 4th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC). Vol. 1. IEEE. 2019, pp. 79–84.

[13] Raia Hadsell, Sumit Chopra, and Yann LeCun. “Dimensionality reduction by learning an invariant mapping”. In: *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR’06)*. Vol. 2. IEEE. 2006, pp. 1735–1742.

[14] Byung-Gil Han et al. “Passenger management system based on face recognition for intelligent transport vehicles”. In: *2013 Fifth International Conference on Ubiquitous and Future Networks (ICUFN)*. IEEE. 2013, pp. 117–118.

[15] Elad Hoffer and Nir Ailon. “Deep metric learning using triplet network”. In: *International workshop on similarity-based pattern recognition*. Springer. 2015, pp. 84–92.

[16] Forrest N Iandola et al. “SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and¡ 0.5 MB model size”. In: *arXiv preprint arXiv:1602.07360* (2016).

[17] The Independent. *Inside Kampala’s Public Transport Politics*. https://www.independent.co.ug/inside-kampalas-public-transport-politics/. [Online; accessed 31-January-2021]. 2020.

[18] Intel. *face-recognition-mobilefacenet-arcface*. https://docs.openvino.ai/2021.2/omz_models_public_face_recognition_mobilefacenet_arcface_face_recognition_mobilefacenet_arcface.html. [Online; accessed 27-October-2021].

[19] Garima Jain and Sanjeet Dahiya. “NFC?: Advantages, Limits and Future Scope”. In: *vol* 4 (2015), pp. 1–12.

[20] Ari Juels. “RFID security and privacy: A research survey”. In: *IEEE journal on selected areas in communications* 24.2 (2006), pp. 381–394.

[21] Kivy. *Kivy Python Framework*. https://kivy.org. [Online; accessed 27-October-2021].

[22] Gregory Koch, Richard Zemel, Ruslan Salakhutdinov, et al. “Siamese neural networks for one-shot image recognition”. In: *ICML deep learning workshop*. Vol. 2. Lille. 2015, p. 0.

[23] Izabela Lacmanović, Biljana Radulović, and Dejan Lacmanović. “Contactless payment systems based on RFID technology”. In: *The 33rd International Convention MIPRO*. IEEE. 2010, pp. 1114–1119.

[24] Saibal Manna, Sushil Ghildiyal, and Kishankumar Bhimani. “Face recognition from video using deep learning”. In: *2020 5th International Conference on Communication and Electronics Systems (ICCES)*. IEEE. 2020, pp. 1101–1106.

[25] Collin Mulliner. “Vulnerability analysis and attacks on NFC-enabled mobile phones”. In: *2009 International Conference on Availability, Reliability and Security*. IEEE. 2009, pp. 695–700.

[26] Surya Michrandi Nasution, Emir Mauludi Husni, and Aciek Ida Wuryandari. “Prototype of train ticketing application using Near Field Communication (NFC) technology on Android device”. In: *2012 International Conference on System Engineering and Technology (ICSET)*. IEEE. 2012, pp. 1–6.

[27] Kaley Overstreet. *Bus or Bust? The Future of Public Transit in Life After COVID-19*. https://www.archdaily.com/952441/bus-or-bust-the-future-of-public-transit-in-life-after-covid-19. [Online; accessed 31-January-2021]. 2020.

[28] NG Partha Sarathi and C Obed Otto. “Fare management system for transport corporation using face recognition based on principal component analysis”. In: *2014 International Conference on Green Computing Communication and Electrical Engineering (ICGCCEE)*. IEEE. 2014, pp. 1–5.

[29] Florian Schroff, Dmitry Kalenichenko, and James Philbin. “Facenet: A unified embedding for face recognition and clustering”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015, pp. 815–823.

[30] SolidWorks. *SolidWorks software*. https://www.solidworks.com. [Online; accessed 27-October-2021].

[31] Sandeep Tamrakar, Jan-Erik Ekberg, and N Asokan. “Identity verification schemes for public transport ticketing with NFC phones”. In: *Proceedings of the sixth ACM workshop on Scalable trusted computing*. 2011, pp. 37–48.

[32] Roel Verdult and François Kooman. “Practical attacks on NFC enabled cell phones”. In: *2011 Third international workshop on near field communication*. IEEE. 2011, pp. 77–82.

[33] Ron Weinstein. “RFID: a technical overview and its application to the enterprise”. In: *IT professional* 7.3 (2005), pp. 27–33.

[34] RF Wireless World. *Advantages of NFC — disadvantages of NFC*. https://www.rfwireless-world.com/Terminology/Advantages-and-Disadvantages-of-NFC.html. [Online; accessed 31-January-2021]. 2020.

[35] Maheen Zulfiqar et al. “Deep face recognition for biometric authentication”. In: *2019 International Conference on Electrical, Communication, and Computer Engineering (ICECCE)*. IEEE. 2019, pp. 1–6.