A Synopsis on

Using AI for Designing ID Cards Embedded with Invisible QR Code

Submitted in partial fulfillment of the requirements
of the degree of

Bachelor of Engineering

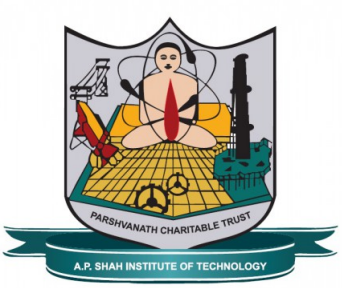
in

Information Technology

by

Chirag Jain 18104002
Dhruva Mhatre 18104045
Prem Vispute 18104059

Guide
Prof. Kaushiki Upadhyaya



Department of Information Technology
A.P. Shah Institute of Technology
G.B.Road,Kasarvadavli, Thane(W), Mumbai-400615
UNIVERSITY OF MUMBAI
2021-2022

CERTIFICATE

This is to certify that the project Synopsis entitled ***"Using AI for Designing ID Cards Embedded with Invisible QR Code"*** Submitted by ***"Chirag Jain (18104002), Dhruva Mhatre (18104045), Prem Vispute (18104059)"*** for the partial fulfillment of the requirement for award of a degree ***Bachelor of Engineering*** in .to the University of Mumbai,is a bonafide work carried out during academic year 2021-2022

Prof. Kaushiki Upadhyaya
Guide

Prof. Kiran Deshpande
Head Department of Information Technology

Dr. Uttam D.Kolekar
Principal

External Examiner(s)

1.

2.

Place:A.P.Shah Institute of Technology, Thane
Date:

Declaration

I declare that this written submission represents my ideas in my own words and where others' ideas or words have been included, I have adequately cited and referenced the original sources. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. I understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

_____

(Signature)

Chirag Jain 18104002

Dhruva Mhatre 18104045

Prem Vispute 18104059

Date:

Abstract

Convolution neural networks (CNN) is a class of Deep neural networks that provides us with the greater capabilities for understanding visual imagery and recognising Image patterns. Here we aim to use CNN to embed an arbitrary image (QR code image) into a colour image. QR (Quick Response) codes are very effective for hiding arbitrary information and are widely known for their error correcting mechanism. Here we aim to hide arbitrary information into a QR code and then embed that QR code into an arbitrary image (colour image). For achieving this goal, we propose a model consisting of two CNNs (encoder CNN decoder CNN). The role of encoder CNN will be to embed the QR image(I1) into the colour image(I2), such that the output of it will be the coloured image (O) embedded with the QR image (Invisible). The role of the decoder CNN will be to take the coloured image (O) as input and restore the embedded QR image (I1). Our system prevents the duplication of the general QR code and thus prevents its misuse by providing better security and hiding important information.

Keywords

Convolution neural network, Deep Learning, QR code, Steganography

Introduction

Identification cards or Id cards are the most important credential for any organization and it helps uniquely identify the person or verify them. But these ID cards may contain some personal information like the mobile number or the address of the person. Such a crucial information, if fallen into wrong hand could be misused. Hence, here we are proposing a system where all the information of the user will be stored in a QR code and that QR will be embedded in the user's image such that it will be invisible to the naked eyes. The main aim of our project is to develop a system which will embed the QR code (containing arbitrary user information) into a coloured image (User photograph), such that the QR image will be hidden in the background and the main visible image will be the user image [1]. For developing this system, we are using the methodology of convolution neural networks.

Convolution neural networks (known as ConvNet or CNN) is a branch of Deep learning that is commonly applied to analysing visual imagery. CNNs are proven to be effective in the areas of image classification and recognition. A CNN model works by extracting necessary features from images in the form of pixels, which helps in reducing the image size and provides us with data

which is important in image recognition. CNN have multiple layers; including convolutional layer, non-linearity layer, pooling layer and fully-connected layer [2]. When an image is given as an input to a CNN trained model, the first layer that acts on it is the convolution layer. This layer consists of a filter/feature detector (of specified size) which moves all over the image matrix, multiplying its values with the original pixel values. The main purpose of this layer is to provide a matrix consisting of extracted features which is much lesser in size. After every convolution layer a non-linearity layer is added. This layer uses activation function to bring non linearity to the data such that a complex network is created to extract new patterns in the image. Pooling layer is generally used to further downsize the image by extracting more important pixels but using certain functions like Max pooling, Min pooling etc. This helps in reducing the training time of the network by reducing the number of computations required. In fully connected layer each and every neuron from the previous layer is connected to every neuron of the next layer. This is similar to the traditional arrangements of neural networks.
For our system, we are developing a model consisting of two CNNs; encoder CNN decoder CNN. The role of encoder CNN will be to take an input of the QR code image and colour image, and generate an output of coloured image embedded with the QR code image. The role of decoder CNN is to recover the original QR code image from the coloured image [1].

Literature Review

Steganography is a method of hiding important data in an unsuspicious media such as text, audio, video or image in such a way that there is no knowledge of the existence of the information in the first place. Most popular form of steganography used nowadays is using images as source of hiding data. These methods involve hiding message by altering "noisy" areas with using methods like least-significant bit or LSB, which directly alters the image pixels value [3]. This method of image steganography limits the capacity of data that can be hidden in the carrier image, if more data is added it may distort the image leading to a significant difference between the original image and the stego-image. To overcome this drawback, the authors of [4] proposed a method for image steganography using Fully Convolutional Dense Network (FC-DenseNet).
Convolution neural networks are easier to train if they have shorter connections, and perhaps produce more accurate and efficient results. Dense convolutional neural networks (hereafter referred as DenseNet) connects each layer to every other layer in a feed-forward fashion [5]. DenseNets can be considered as an extension of ResNets [6], where each layer obtains additional inputs from all preceding layers and passes on its feature-maps to all subsequent layer, where in Resnet we combine features through Summation where as in DenseNet features are combined through the method of concatenation before passing into another layer [5]. FC-DenseNets is mainly used for semantic segmentation. It uses an upsampling layer for recovering the spatial resolution of the input or output layer. Only the feature maps of preceding dense blocks are upsampled in order have number of dense blocks at each resolution of the upsampling path independent of number of pooling layers. The upsampled dense blocks combines the

information of other dense blocks of same resolution and the higher resolution information is by the standard skip connection between the upsampling and the down sampling paths [7]. We are using FC-DenseNets for hiding the image as shown in [4]. Here the authors are hiding information (secret image) into a carrier image in such a way that the stego-image will be visually similar to the carrier image. In our project we are using this FC-DenseNet method for hiding a QR code image into a carrier image as shown in [1]. We are generating a QR code consisting of user data (such as phone no., address, email etc.), and are embedding the QR code image into the user image. The goal of this system will be to provide data hiding through Cryptography and steganography, such that the user data will be hidden through QR generation and hiding the data through Steganography so that there will be no knowledge of the existence of the data in the first place.

System Architecture/Working

The system in this project will be developed in order to provide steganographic mechanisms by hiding the QR image (consisting user data) into the User's photograph (carrier image) such that the QR image will be invisible to the naked eyes. The entire system will mainly consist of QR code generator for generating QR code image consisting of arbitrary user information, an encoder CNN for hiding the QR code image into the user image, and an Decoder CNN for extracting the QR code from the User embedded image.

1. QR Code Generator:

QR code or the Quick Response code are machine-readable two-dimensional barcodes which can store a variety of information and are widely known and used for its error correcting mechanism. The QR code generator model of our project is responsible for generating a QR code by taking user information as input. Here user data such as name, number, address etc. will be provided to the system, and as an output the system will generate an QR code encrypting the given information as shown in the figure. We are using pyqrcode library of python to develop this QR code generator.

2. Encoder CNN:

The role of encoder CNN is to embed the QR image generated in section 1. into the user image such that the QR image will not be visible to the naked eyes and the output image of encoder CNN will be visually similar to the user image. The QR image and the User image will be the input to the encoder CNN and the output to it will be an Image O which will be similar to the user image but will consist of the QR code image hidden in the background. To develop the encoder CNN, we are using a model of Fully – Connected DenseNet [4]. Encoder components include two 3×3 convolution, 11 Dense Block (DB), 10 Concatenation, 5 Transition Down (TD) and 5 Transition Up (TU). After going through a series of processes several times, the functionality of both images is fully combined and the embedded image is output.

3. Decoder CNN:

The role Decoder CNN is to extract the QR code image from the image O. It takes the output image of the encoder CNN as input and provides us with the original QR image which was embedded in the image O. r. The decoder components include 6 convolutions, 5 batch normalization (BN), and 6 Re LU.

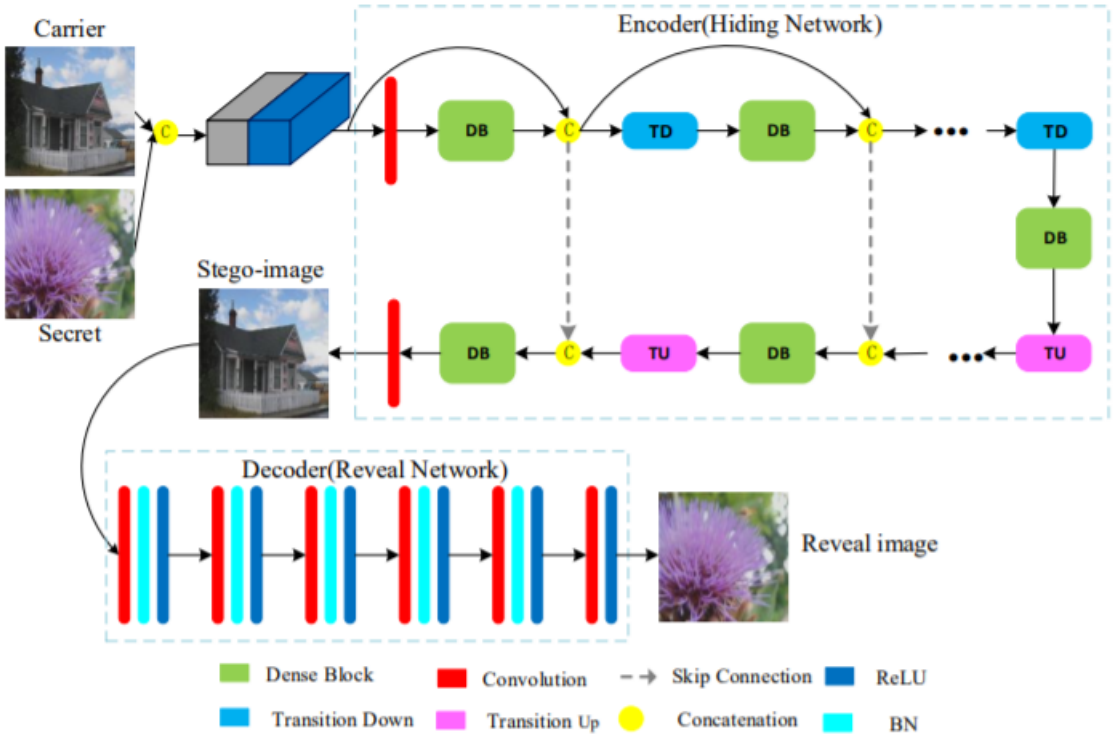


Illustration of encoder decoder model [4]

References

[1] K. Yamauchi and H. Kobayashi, "Invisible QR Code Generator Using Convolutional Neural Network," IECON 2020 The 46th Annual Conference of the IEEE Industrial Electronics Society, 2020, pp. 4009-4014, doi: 10.1109/IECON43393.2020.9254709.

[2] S. Albawi, T. A. Mohammed and S. Al-Zawi, "Understanding of a convolutional neural network," 2017 International Conference on Engineering and Technology (ICET), 2017, pp. 1-6, doi: 10.1109/ICEngTechnol.2017.8308186.

[3] Steganography and Steganalysis J.R. Krenn January 2004

[4] Hide the Image in FC-DenseNets to another Image Duan Xintao, Liu Nao

[5] Densely Connected Convolutional Networks Gao Huang, Zhuang Liu, Laurens van der Maaten, Kilian Q. Weinberger

[6] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. CoRR, abs/1512.03385, 2015.

[7] The One Hundred Layers Tiramisu: Fully Convolutional DenseNets for Semantic Segmentation Simon Jégou, Michal Drozdzal, David Vazquez, Adriana Romero, Yoshua Bengio