PRE-PROGRAMMED WEB PAGE IMPLEMENTATION FOR MOCKUP IMAGE

JonnadulaNarasimha Rao, *Research Scholar, Department of CSE, Madhav University, Rajasthan, India, Associate Professor, CMR Technical Campus.*

GajulaHarish, *Annapurna das, Y.Tejasvi,Department of CSE, CMR Technical Campus*

SECTION I

Abstract

The design of a website is generated by creating mock-ups either by hand or using graphic design and mock-up tools. In this project, when the mock-up image is uploaded to the server, the image details are stored in MYSQL database. Now the image will undergo object detection and produce morphological transformation i.e., dilation, erosion, threshold, greyscale. After the threshold process, rectangles were drawn by applying the counter detection algorithm to determine the objects. In this way, the components in the input image have been detected. The detected componentsare cropped to be transformed.Now we will train the model with respect to the mock-up image and it is converted into structured HTML by software engineers. Hand drawn mock-ups are processed using deep learning methods to replace from existing to proposed model.

Introduction

As the technology increases, there are various things which are progressed with it. One amongst them is website. A website is nothing but a page over internet which tells specifically about a certain matter. Every current running business, schools, organizations, companies, etc., have their own websites. They are used for sales, marketing, advertising, products and many more. They are even used by companies for financial reasons.

Every user is attracted by the way webpage is created;it is important to create curiosity within user. To create a webpage different people from different fields are required to team up together. The procedure starts by creating mock up images, either on paper or graphically. Based on this version and depending on the mock up image made, the code is created by experts. The code is written by software experts to fulfil the requirements of the mock ups. This process involves constant functioning of task. The end webpage will then be visible to end users and changes are done based on their reaction. Rewriting the code and changing structure of page is tough task. The experts must research for more effective results to design webpages.

The user reviews and growth in technology gives us the idea of generating automatic code for webpage. This can reduce the cost of processing, time to program, resources requirementsand complexity of code. By using these design states, the website is created within shorter time.
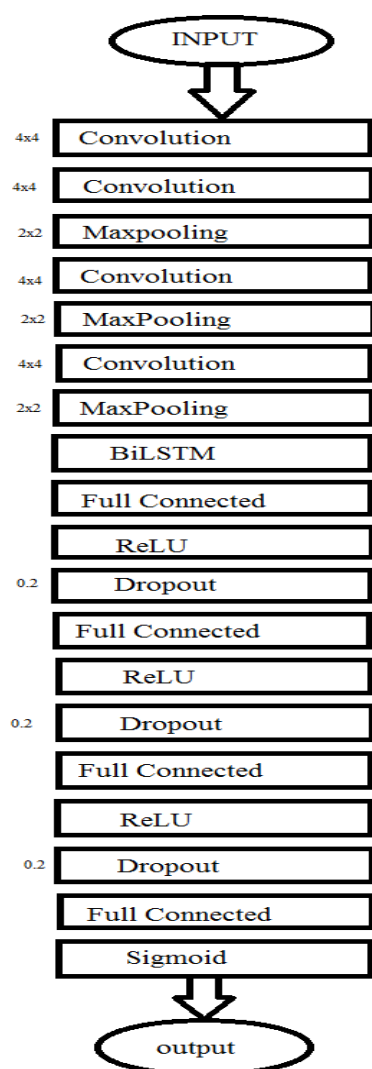
This project is aimed to recognize components created in mock-up and encode them according to webpages hierarchy, this can be done by an algorithm which has been developed in this to create pre-programmed HTML code. The proposed layout is obtained from public datasets of hand drawn images. With the help of computer vision techniques and convolution neural network, the datasets are trained and processed. At the final level, HTML code is obtained.

The rest of the paper is followed as; Section II tells about the algorithm and model. Section III and IV describes the methods or modules and dataset respectively. Section V gives the results. Section VI gives the conclusions weobtained.

Section II

Algorithm and related work

The algorithm used in this project is REMAUI that is, Reverse Engineering Mobile Application User Interface. It is used to recognize the GUI components like buttons, textboxes and creates code for them from the screenshot of the mock up image. In our study, this was the first one where in we used recognition methods like, computer vision and optical character. Animated images are not allowed because we cannot hand draw them and transaction of cross pages are also not present. But in spite of these drawbacks also, REMAUI works successfully in creating the code for the screenshots or graphical images.

	INPUT
4x4	Convolution
4x4	Convolution
2x2	Maxpooling
4x4	Convolution
2x2	MaxPooling
4x4	Convolution
2x2	MaxPooling
	BiLSTM
	Full Connected
	ReLU
0.2	Dropout
	Full Connected
	ReLU
0.2	Dropout
	Full Connected
	ReLU
0.2	Dropout
	Full Connected
	Sigmoid
	output

CNN Model

To replace the drawbacks of REMAUI, authors created other algorithms. One amongst them is P2A algorithm, which fulfils the drawbacks of REMAUI. There are various methods and techniques to convert graphical image to code. The pix2code is one amongst them which can convert the graphical user interface to code in structure. It also uses convolution neural network model. This technique was verified on different applications and passed successfully.

These days, codes for all kinds of applications are available through opensource. An algorithm named ReDraw is used here with KNN algorithm. Structured HTML code is created by using this combination of algorithm. Firstly, the computer vision techniques are utilized and the components of graphical user interface are detected. Next, the components are classified according to their functions which can be buttons, text-box, etc. Lastly, the HTML code is generated using the KNN algorithm according to web page hierarchy. Deep convolution neural networks are also used.

The codes generated by experts are stored in open sources so that it can be reused for practice or to start new project. Different people use this saved codes and improvise it and reduce the complexity and update it again. For example, authors use search program called CALL in which user gives graphical images and keywords. This interface is then searched in existing libraries to match the same and obtain the code and content. End user gets all the related options and finally chooses one amongst them. Microsoft implemented design with custom vision model using AI tools that creates structured HTML code using mock ups.

SECTION III

MethodsorModules

For any process of producing automatic HTML code, there are four basic steps to follow. First the object detection, in this the image is scanned and processed using image processing techniques such as greyscale, erosion, etc. Then the identified objects are cropped. Finally, the output of the model is converted into HTML code.

The object detection and pruning, object recognition, sketch modules are bundled inside the Mock-up Image Pre-processing.

A OBJECT DETECTION AND PRUNING(cropping)

The image uploaded is the input file, after reading it is converted to greyscale format. Then the Gaussian function is applied to reduce the noise. Then meeting the requirements of the threshold process, rectangles are drawn using counter detection process for shape analysis. They are converted into morphological transformations. The detected components are cropped and transferred to the CNN model. In thisstage, morphological transformations are performed which undergo dilation and erosion process.
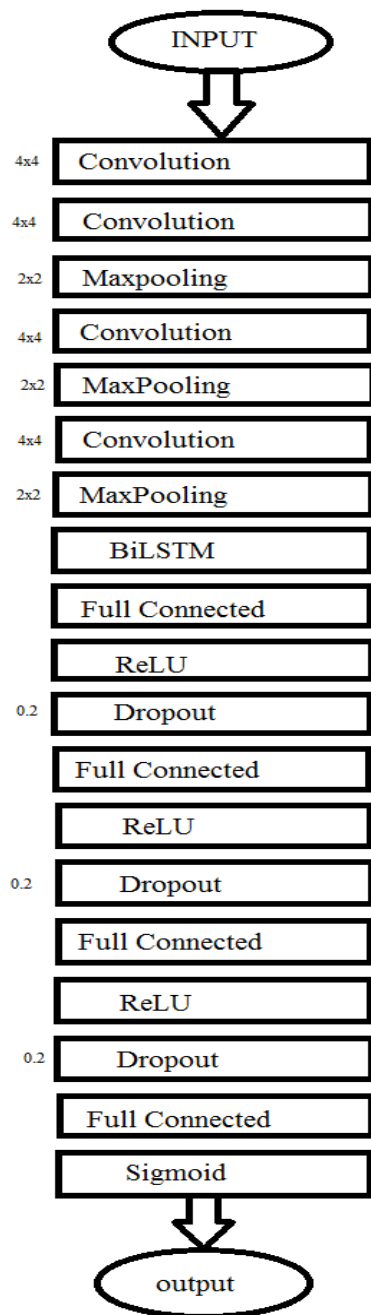
B. OBJECT RECOGNITION

The recognised components such as text box, drop down, button and checkbox are trained with dataset.

After this stage, the model is trained with loss function to avoid deviation from actual results. Then components that come as input to this stage from previous stage are carried out with component recognition process. As we see in the model below, the layers enhance the importance in the image to be able to differentiate one from the other. We keep several convolution layers, with 4*4 kernels and apply max pooling for feature extraction. Then we keep BiLSTM layer to catch correlation of the extracted features. This process is called vectorization. Later we use Full connected layers and Dropout layers with 20% accuracy in order to achieve objective to classify.

INPUT

4x4 | Convolution

4x4 | Convolution

2x2 | Maxpooling

4x4 | Convolution

2x2 | MaxPooling

4x4 | Convolution

2x2 | MaxPooling

BiLSTM

Full Connected

ReLU

0.2 | Dropout

Full Connected

ReLU

0.2 | Dropout

Full Connected

ReLU

0.2 | Dropout

Full Connected

Sigmoid

output

CNN Model

C. SKETCH

It enhances the image in a better way. It uses image captioning to generate HTML code. We can generate colourful text boxes and buttonsto attract the end user through output on webpage. It uses pix2code algorithm to generate the output synthetically.
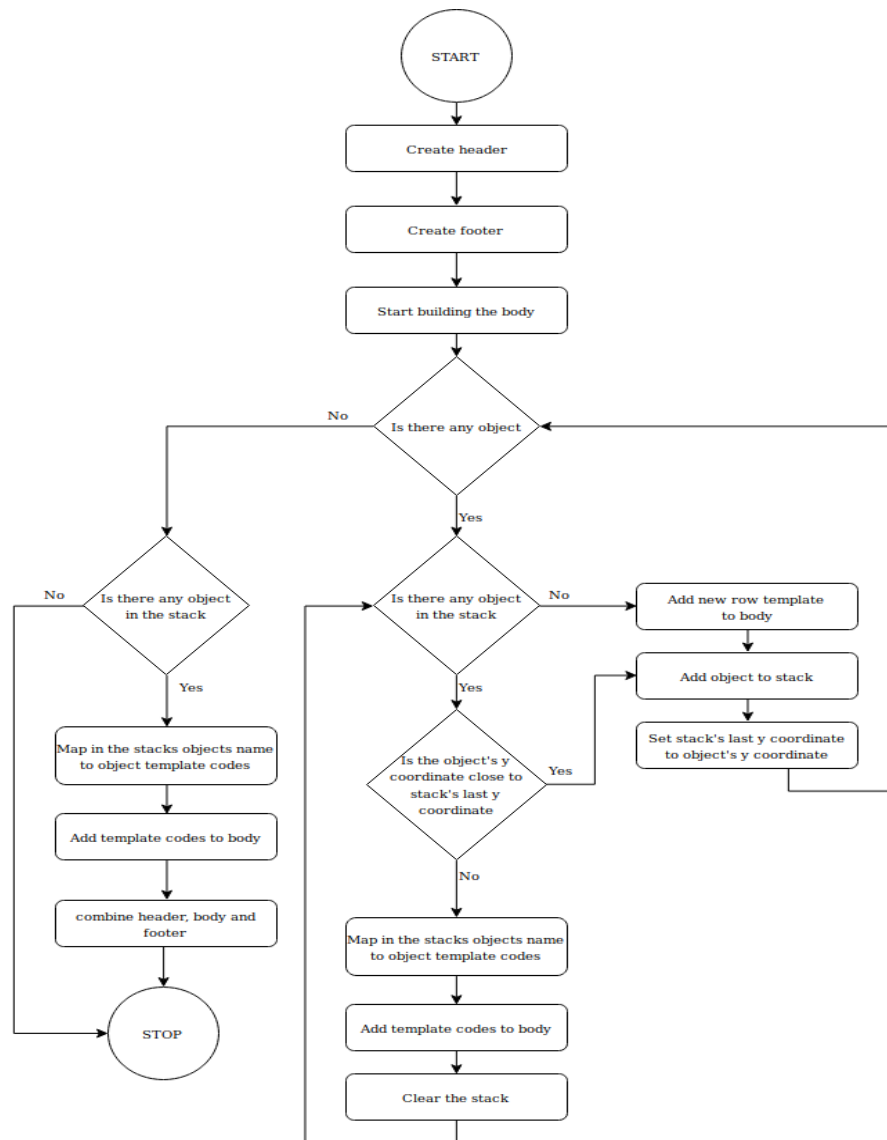

D. HTML BUILDER

The output of the image is obtained in this stage. The components which are recognised are translated to HTML code through bootstrap framework. As we see in the HTML builder

algorithm, firstly we created templates. Next, we detect items on each row with coordinates of components. Then we map it with template code. At the end the code is generated successfully. All the templates are combined together and executed in the output page. The aim of the project to transform image to code is done successfully.

OUTPUT:
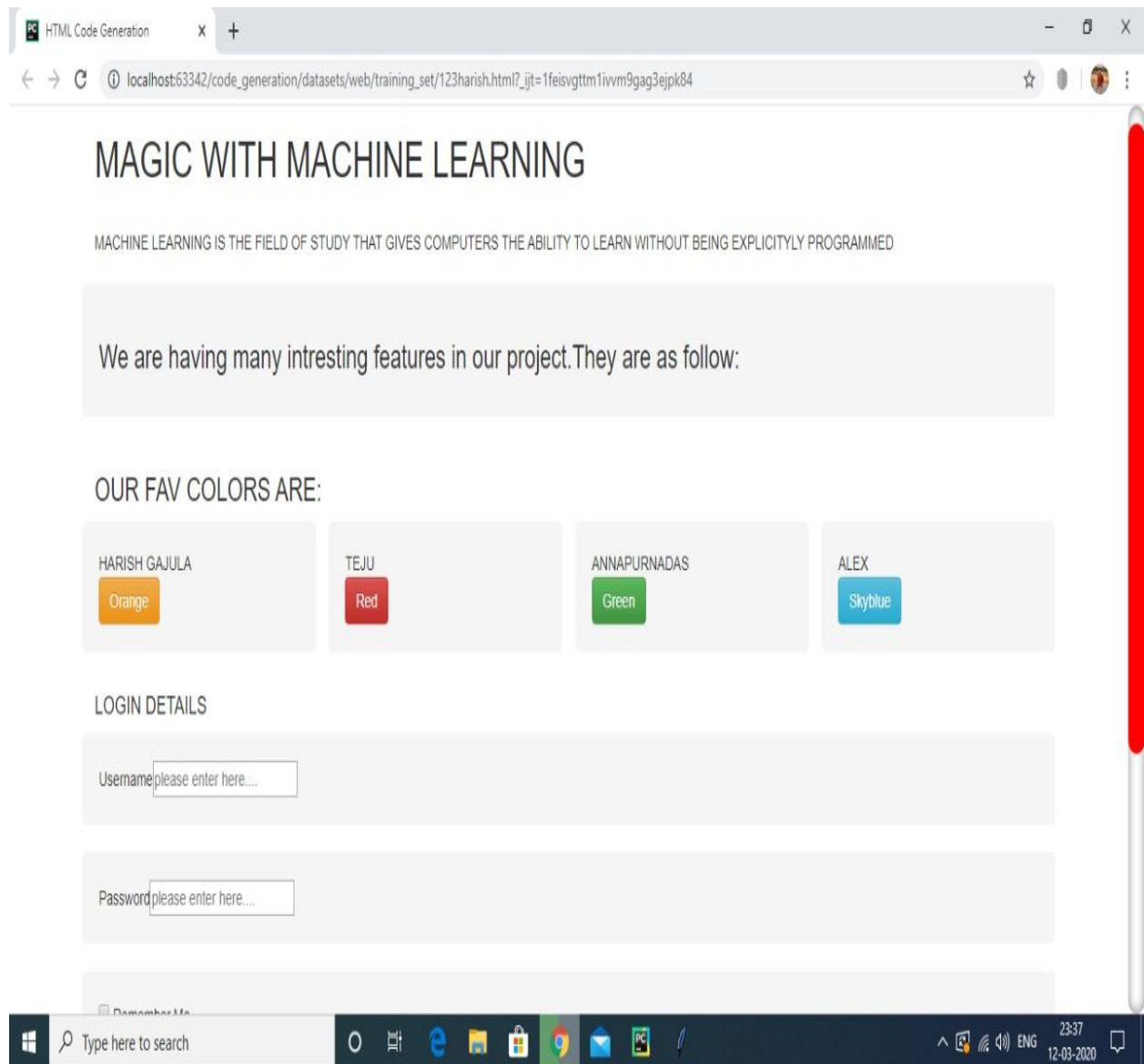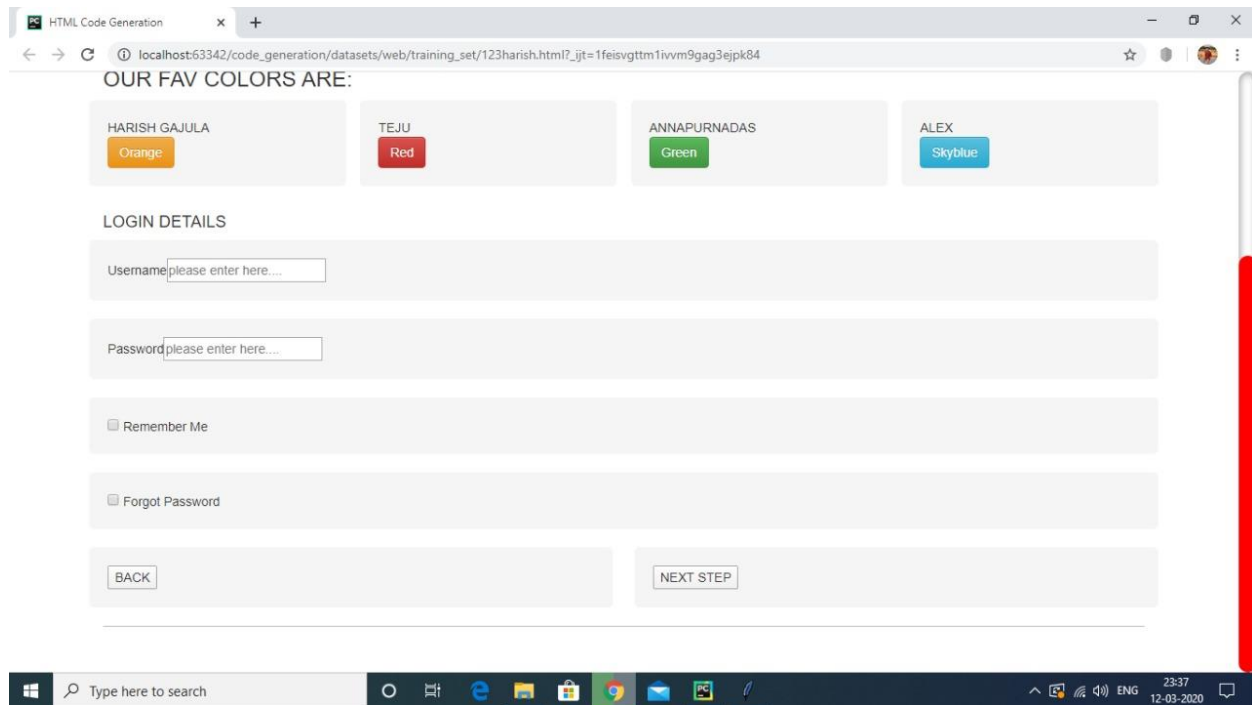
SECTION IV

RESULTS

In this study, the purpose is implemented successfully. We succeeded in generating the HTML code from mock up image with extra feature of colour buttons. All the methods are the base elements which made this existing to proposed system. The morphological transformations also played a key role. Our results show 90% accuracy and validation of the project.

SECTION V

CONCLUSION

The era of using programming languages is now wrapped up. Artificial intelligence is entering into every field and is giving success to every filed in industry. Converting the mock up into HTML code within less time and less efforts is significant achievement in current years. At the end, we have a dataset that consists of our mock up images. We developed a system that can automatically give the HTML code. The dataset was created using all different components

present and was used to train CNN model to recognise the object. As a result with 90% accuracy we successfully trained the model and enabled the code out of it.

REFERENCES

[1] Sketch2code.Microsoft AI Labs. [Online]. Available: https://github.com/ Microsoft/ailab/tree/master/Sketch2Code/model/images

[2] S.Natarajan and C.Csallner"P2A: A Toolfor Converting Pixels to Animated Mobile Application User Interfaces," *Proceedings of the 5th International Conference on Mobile Software Engineering and Systems-MOBILESoft'18*, pp. 224–235, 2018. [Online]. Available: http://dl.acm.org/citation.cfm?doid=3197231.3197249

[3] [Online]. Available: https://github.com/

[4] S. P. Reiss, Y. Miao, and Q. Xin, "Seeking the user interface,"*Automated Software Engineering*, vol. 25, no. 1, pp. 157–193, mar 2018. [Online]. Available: https://doi.org/10.1007/s10515-017-0216-3