GENERATION OF HTML CODE AUTOMATICALLY USING MOCK-UP IMAGES WITH MACHINE LEARNING TECHNIQUE

Mr. Pritish Dinesh Sathe[1], Mr. Parth Archana Kapile[2], Mr. Tarun Birjuchand Uike[3], Ms. Tanisha Motiram Wankhade[4], Prof. Neeraj Sahu[5]

[1-4]Student of Computer Science of Engineering Department,G. H. Raisoni University of Amravati Maharashtra
[5]Professor, Dept. of Computer Science and Engineering, G. H. Raiosni University Amravati, Maharashtra, India

---***---

Abstract - *The initial step in website design is to create mock-up images for certain web pages, either by hand or with the use of mock-up development tools. It is effectively used by developers to move mock-ups of web pages to coding. It's developing the wireframe for the layout interfaces and generating the proposed system. Computer vision and deep systematic analysis are the two most commonly employed techniques. The use of automatic code creation saves time and money. We've organized and sketched out the design. This design is cost-effective and time saving in terms of development of any website.*

Key Words: **HTML Code, Machine Learning, Deep Learning, Automatic HTML Generation, CNN**

1. INTRODUCTION

Everyone in today's world understands the value of the internet. The internet and websites have become an inextricable aspect of our lives. Data, as we all know, is the new oil in today's market. The most crucial component is the website. Websites are a highly popular and significant aspect in all types of businesses. Websites are used for a variety of purposes, including education, knowledge, entertainment, and social interaction. The designing phase of a website is the first step. The initial step in the building of a website is to design its appearance. It can be done by drafting the design by hand or using any software to create mock-up images. The team of web developers will then convert these mock-ups into HTML code. This process continues until the intended outcomes are achieved. Our major goal in this project is to build HTML code using these photographs (images). To accomplish this, we will employ convolutional neural networks, computer vision technology, and deep learning approaches. Any website's front end is concerned with the site's users. It's important to elicit a particular level of interest from the user. It is simple to use and contains sufficient sophisticated features. Creating a webpage that provides active responses expertly, on the other hand, necessitated a long and exhausting journey. Many web developers or software specialists collaborate throughout the development of a website or a specific page to create an accurate and appealing front end. On the basis of the draught, the developers create the code to design the webpage. The generated webpage, on the other hand, can be modified based on the user's response. As a result, altering the structure to meet the needs of the user is a challenging task. These requirements necessitate the inclusion of improved functionalities to a website format. The concept of arranging a webpage with automated code is an intriguing research topic. Automatic webpage generation saves the time, money and resources by reducing coding time, all the tedious processes and obviously the cost. As a result of this quick pattern, the ultimate product, a website, is created in a relatively short time. In this study, we used methods for autonomously generating hand drawn graphics by generating code. Its purpose is to decrypt the system in a website format and examine the element that creates the hand drawn graphics.

2. PROPOSED MODEL

The major purpose here is to transform the manually drawn mock-ups into an HTML code to make a template for a website or for a front end of the website according to as it is drawn. The mock-up which contains button, text field etc., we are aiming to convert them into an HTML code. In order to transform them into HTML code to frontend template we need to use CNN model, object recognition, cropping, dilution, erosion (computer vision techniques). The work is to use object detection algorithm to detect the component from the manually drawn images. The work is to use object cropping algorithm to crop an object like text field, radio button, button etc. The object recognition algorithm works on Convolution Neural Network (CNN) to train the purposed system by using data set and it works successfully. The next work is of HTML builder algorithm and that is to convert the item which is detected by object recognition algorithm using Convolution Neural Network (CNN) model to an HTML code using HTML code builder algorithm.

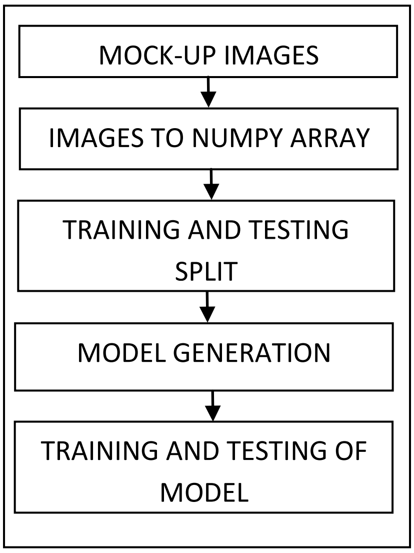

Fig: WORKFLOW OF THE SYSTEM

When employing an HTML builder algorithm, the first step is to master Computer Vision (cv) techniques, which are needed to recognize each and every Graphical User Interface (GUI) element. The second stage is to classify the components based on their workstyle, such as textbox as textbox, button as button, checkbox as checkbox, and so forth. We employ a Convolutional Neural Network (CNN) for this procedure, and the resulting Extensible Markup Language (XML) code is formatted according to web programming conventions.

2.1 DATASET

To construct our dataset in the suggested method, we used hand drawn mock-ups. We drawn an image containing textboxes, button, radio buttons, checkboxes and other elements. Following that we sent it to our CNN model to train.

2.2 OBJECT DETECTION

The picture is translated into an array format after receiving an input file in image format from the user. The 3*3 rectangle kernel was built to identify them in this manner. After that, the pieces are clipped and passed to the Convolution Neural Network Model for the next step.

2.3 OBJECT RECOGNIITON

With the help of components from our dataset, the model was able to transfer. Checkboxes, textboxes, radio buttons, and other components make comprise a dataset. The components are sent to the Convolutional Neural Network (CNN) model for component recognition. For object recognition there will be different techniques that will be used from Computer Vision (CV) library.

2.4 CONVULUTION NEURAL NETWORK (CNN)

The convolutional layer, the pooling layer, the ReLU layer, and the fully connected layer are the four layers in the CNN model. The first block performs the function of a feature extractor. It uses convolution filtering processes to accomplish template matching. The first layer applies numerous convolution kernels to the picture and returns featured maps. The pooling layer decreases the size of the image while preserving its essential properties. It also minimizes the number of calculations, improving the network's efficiency. The ReLU layer replaces all negative values with zero in the third layer. To make the CNN operate, all of the levels work together. The whole procedure in our project is completed by directly importing the Keras library.

2.5 HTML BUILD

Following the completion of the recognition process, the detected items are successfully transformed to HTML code utilizing the framework. The coordinates from the counter finding algorithm output are used to finish this procedure. First, we developed templates for the website's header and footer using the HTML builder algorithm. The next step is to recognize the number of items in each row and their coordinates. Then we just labelled each element's code for their relevant template code. After that, HTML code for the body section was created. Finally, the footer and header were concatenated to create the whole HTML code.

2.5 COMPUTER VISION

Object detection and recognition are both covered by the Computer Vision library, often known as OpenCV in Python. ImageAI, Single Shot Detectors, You Only Look Once (YOLO), and Region Based Convolutional Networks are some of the most often utilized algorithms in OpenCV. To use the library, you must first install the dependencies. In our project, we used tensorflow libraries.

2.6 METHODS USED

In order to accomplish this study, we followed four phases. The first step is to perform an object detection method on the incoming data. Following that, using the tensorflow libraries, the pieces that were recognized are cropped. The final result of this method is converted into HTML code via an HTML build algorithm.

3. ADVANTAGES OF THE SYSTEM

1. Quick front end development of the website.
2. Cost Effective. Reduces the cost of development.
3. Easy to implement and process images.

4. DISADVANTAGES OF THE SYSTEM

1. Attractive designs are difficult to implement using this system.
2. Can cause difficulties while detecting some GUI elements.
3. Works only when the provided input are images.

5. RESULTS

As a consequence, the goal of automatically creating HTML code using mock-ups was achieved. This procedure was implemented in four phases, each of which was completed one at a time. Exposure of an item, modification of an item, accepting the item, and HTML builder were among the steps. These processes were carried out in the order that they were intended to be carried out. In the object identification stage, a Convolution Neural Network (CNN) layered model was utilized. Finally, a new method for HTML code propagation was awarded, which is the culmination of the overall intended structure.

6. CONCLUSION

An important element has been the transformation of mock-ups into HTML code in less time and with lower development costs. We built a technique in this study that receives mock-up pictures of web pages, performs the needed actions on them, and generates the desired HTML build. A data collection containing photos containing various mock-ups of web page architecture was successfully used. The Convolution Neural Network (CNN) model is then trained using the dataset. We predicted what is possible with this project. Our model was created using a smaller dataset, yet it was still successful. The nature of the model may undoubtedly be improved by creating a larger model and including more data over a longer time period.

REFERENCES

[1] A Graves. Generating Sequences with Recurrent Neural Network.arXiv preprint arXiv:1308.0850,2013

[2] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. Distributed Representation of Words and Phrases and their Compositionality. In Advance in Neural Information Processing Systems, 2013

[3] K. P Moran, C. Bernal- C´ardenas, M. Curcio, R. Bonnet and D. Poshyvaynk, “Machine Learning Based Prototyping of Graphical User Interfaces for Mobile Apps,” IEEE transaction on Software Engineering, 2018

[4] S. Hochreiter and J. Schmidhuber. Long-Short Term Memory. Neural Computation, 1997.

[5] A. Karpathy and L. Fei Fei. Deep Visual-Semantic Alignments for Generating Image Descriptions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015.

[6] [Online] Available: https://www.github.com/

[7] Survey and Analysis on Machine Learning Approaches for Image Annotation V. R. Palekar, DS Kumar L Journal of University Shanghai for Science and Technology

[8] Sktech2code. Microsoft AI Labs. [Online] Available: https://github.com/Microsoft/ailab/tree/master/Sketch2Code/model/images

[9] B. Aşıroğlu *et al*., "Automatic HTML Code Generation from Mock-Up Images Using Machine Learning Techniques," *2019 Scientific Meeting on Electrical-Electronics & Biomedical Engineering and Computer Science (EBBT)*, 2019, pp. 1-4, doi: 10.1109/EBBT.2019.8741736.

[10] Aşiroğlu, Batuhan & Mete, Busta & Yildiz, Eyyup & Nalçakan, Yağız & Sezen, Alper & Dagtekin, Mustafa & Ensari, Tolga. (2019). Automatic HTML Code Generation from Mock-Up Images Using Machine Learning Techniques. 1-4. 10.1109/EBBT.2019.8741736.

[11] Journal, IRJET, and Yağız Nalçakan. “IRJET- Automatic HTML Code Generation from Mock-Up Images Using Machine Learning Techniques.” IRJET (2021): n. pag. Web.

[12] Tolga Ensari, Alper Sezen, Mustafa Dagtekin, “Automatic HTML Code Generation from Mock-up Images Using Machine Learning Techniques”, IEEE, 2019

[13] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional Neural networks. In Advances in neural information processing systems, pages 1097– 1105, 2012.

[14] J. Donahue, L. Anne Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darrell. Long-term recurrent convolutional networks for visual recognition and description. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 2625–2634, 2015.

BIOGRAPHIES

Mr. Pritish Dinesh Sathe
Student of B. Tech G. H. Raisoni University, Amravati

Mr. Parth Archana Kapile
Student of G. H. Raisoni University,
Amravati

Mr. Tarun Birjuchand Uike
Student of G. H. Raisoni University,
Amravati

Ms. Tanisha Motiram Wankhade
Student of G. H. Raisoni University,
Amravati

Prof. Neeraj Sahu
Department of Computer Science and Engineering, G. H. Raisoni University, Amravati