# Artificial Intelligence based Automated HTML Code Generation Tool using Design Mockups

Prof. S. A. Agrawal[1], Swapnil Suryawanshi[2], Vikas Arsude[3], Nilesh Maid[4]

Department of Computer Engineering, Marathwada Mitra Mandal's
Institute of Technology, SPP University, Pune, India.
[1]sanjay.agrawal@mmit.edu.in
[2]swapnil.suryawanshi@mmit.edu.in
[3]vikas.arsude@mmit.edu.in
[4]nilesh.maid@mmit.edu.in

***Abstract*— Usually the design creation begins with the discussion on paper by sharing ideas. Once the idea is shared and proposed output is drawn, this is shared with the development team to implement that in the code. Then developing the HTML code from using the complex techniques is time consuming and requires an experienced developer. This software is built to convert the user created image into the .html code to help making the customized user interface. In this concept it is almost proven that the machine learning can be used to train the pre-defined models. These models are used to interact and reverse engineer the user given data. This concept can be used to generate a code with a better accuracy. Also, the compatibility can be extended to multiple platforms. This evaluation demonstrates that the deep learning approach highlights our classical computer vision approach and it comes to the conclusion that deep learning is the most capable way for the research ahead.**

***Keywords*— Artificial Intelligence, Web development, Deep Learning, Machine Learning, Automation, HTML.**

## I. Introduction

The user-interface development starts with drawing on paper, this includes ideas shared by designers and engineers and they try their best to represent the exact idea that their customer or client is expecting. Whenever a requested or proposed design is requested or shared, it's usually captured via a photograph/image format and then converted manually into a working HTML webpage that works in a web browser. This approach of development requires time and effort and it often slows down the design process.

In this proposed automation technology concept, we will use various applications of machine learning and artificial intelligence. Here it will give a readymade design or prototype at the end of the discussion which can be directly evaluate and further checked by the engineers and data scientists.

## II. Related Work

Previously proposed system states that, image caption generation has been long observed as a link to connect Natural Language Processing and Computer Vision [1]. Conventionally, this problem was solved by deeply hand-crafted systems. The work in the related papers focused on primarily object, attributes and prepositions detection in images and then makes use of possible graphic models to understand associated captions [1]. Once authors signify imaged by using object proposals and attributes, the powerful language parsing models to generate captions are applied then. However, the predefined templates and language grammar restrict the work of the systems above.

Study of the source code has been one of the core research fields of software engineering. E.g. a considerable amount of research has been conducted on finding symmetric parts from source files and mining code fluctuates in source code directories [2]. Source code of proposed software architecture habitually contains auto-generated code, i.e., parser generators output. The active auto-generated code sometimes hampers source code study.

Consider, if we detect code twins from source code containing auto-generated code, a fair number of code twins are found from auto-generated code. Code twins observed from hand-crafted code should be paid no notice because their count is pretty smaller than the code twins found from auto-generated code. That may lead developers to lose some of the crucial researches by twining detection tools. We can also consider mining source code repositories. This system tracks code elements such as classes and methods are through a large amount of iterations. The active auto-generated code may need more time to fetch code elements. The code comments usually represent themselves as they are auto-generated code. So, we can use tools as UNIX GSrep to find and remove the auto-generated codes. However, programmers do delete those code comments early. Checking files one by one manually is the only way to explore and eliminate the auto-generated code excluding code comments.

## III. Proposed System

To improve the efficiency of the web development process by reducing the workload on the front-end development team. The initial development stages require users to draw multiple wireframes before finalising the final design. The finalised wireframe is the input to the system in Image format. The proposed system will take a screenshot of the wireframe as an input to the system, and generate an auto-generated code from that design mock-up and deliver the generated code file as the output to the end-user.

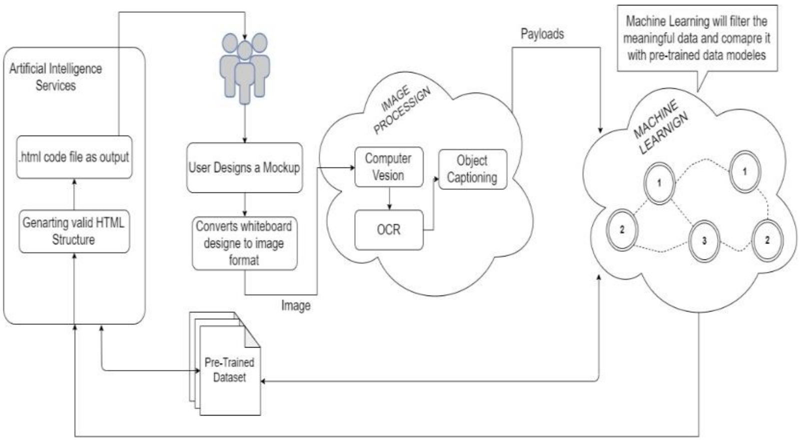
Artificial Intelligence Services
.html code file as output
Genarting valid HTML Structure
User Designs a Mockup
Converts whiteboard designe to image format
Image
Pre-Trained Dataset
IMAGE PROCESSIGN
Computer Vesion
Object Captioning
OCR
Payloads
Machine Learning will filter the meaningful data and comapre it with pre-trained data modeles
MACHINE LEARNIGN
1
1
2
3
2

Fig.1 System Architecture

## IV. System Architecture

Here we are using Deep Learning and Artificial Intelligence services to understand the handwritten mock-ups and translate them into the HTML code. The user will first provide screenshot the design mock-ups as an input to the system. The image processing will extract the meaningful information from the image

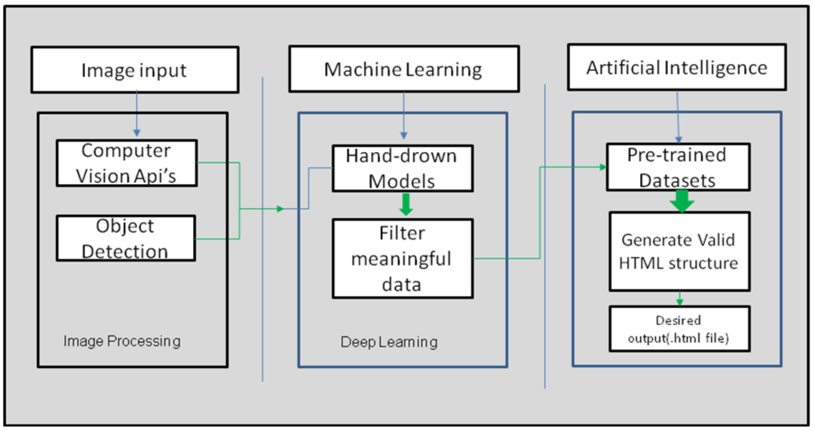

Fig. 2 System Overview

The Artificial Intelligence services will take the output of the previous two steps as an input. Artificial Intelligence will compare the data received with the pre-trained data models. Once the shapes and text are recognized as valid then it generates a structure for creating HTML file. Finally, the HTML code file is generated using the defined attributes by Artificial Intelligence.

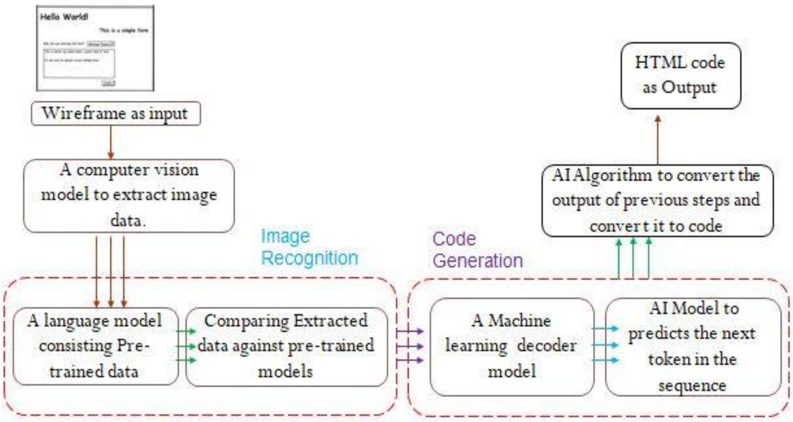
Hello World!
This is a simple form
Wireframe as input
A computer vision model to extract image data.
Image Recognition
Code Generation
HTML code as Output
AI Algorithm to convert the output of previous steps and convert it to code
A language model consisting Pre-trained data
Comparing Extracted data against pre-trained models
A Machine learning decoder model
AI Model to predicts the next token in the sequence

Fig.3 Process Flow Diagram

Here, the two sections for “Imag ing as the input provided. In image recognition the input data is matched with the pre-trained datasets in the storage. If the shapes or texts from the input are not recognized by the system as those are totally different than the shapes of the components used in HTML, the system ignores them and processes only the recognized shapes and text and gives a output for further generation of the HTML structure.

## V. ADVANTAGES

- No need to understand the HTML deeply
- Less time consuming than the manual development
- Easy to design the HTML website
- Easy input and output approach
- Requires less manpower

## VI. CONCLUSIONS

As a beginner’s perspective, earlier it was necessary to have deep knowledge about coding an HTML web page. Building an automated system will surely help and simplify the designing process. By using researches in image captioning and machine learning, this project is able to take hand-drawn design mock-ups and convert them into working HTML code in few seconds.

## ACKNOWLEDGMENT

We would like to express our gratitude to all those who helped us to complete this work. We want to thank our guide Prof. S. A. Agrawal for his continuous help and generous assistance. He helped in a broad range of issues from giving us direction, helping to find the solutions, outlining the requirements and always having the time to see us. We have furthermore to thank Mr. S. G. Rathod, Head of the Department of Computer Engineering to encourage us to go ahead and for continuous guidance. We also want to thank Prof. S. A. Agrawal for all his assistance on Artificial Intelligence based Automated HTML Code Generation Tool using Design Mockups and guidance in research. We would like to thank our colleagues who helped us time to time in preparation research and giving good suggestions. We also extend sincere thanks to all the staff members of Department of Computer Engineering for helping us in various aspects. Last but not least we are grateful to our parents for all their support and encouragement.

## REFERENCES

[1] Senmao Ye, Nian Liu, Junwei Han, Senior Member, IEEE “Attentive Linear Transformation for Image Captioning” JOURNAL OF LATEX CLASS FILES, VOL. 14, NO. 8, AUGUST 2015

[2] Kento Shimonaka∗, Soichi Sumi∗, Yoshiki Higo∗ and Shinji Kusumoto “Identifying Auto-enerated Code by Using Machine Learning Techniques” 2016 7th International Workshop on Empirical Software Engineering in Practice (IWESEP) 13-13 March 2016

[3] Jicheng Fu ; Farokh B. Bastani ; I-ling Yen “Automated AI Planning and Code Pattern Based Code Synthesis” Proceedings of the 18th IEEE International Conference on Tools with Artificial Intelligence (ICTAI'06) 0-7695-2728-0/06

[4] Mert Kilickaya, Aykut Erdem, Nazli Ikizler-Cinbis, and Erkut Erdem “Re-evaluating Automatic Metrics for Image Captioning “arXiv:1612.07600v1 [cs.CL] 22 Dec 2016

[5] MD. Zakir Hossain, Ferdous Sohel, Mohd Fairuz Shiratuddin, Hamid Laga “A Comprehensive Survey of Deep Learning for Image Captioning “(Submitted on 6 Oct 2018 (v1), last revised 14 Oct 2018 (this version, v2))

[6] S. Bengio, O. Vinyals, N. Jaitly, and N. Shazeer, “Scheduled sampling for sequence prediction with recurrent neural networks,” in Advances in Neural Information Processing Systems, 2015, pp. 1171–1179.

[7] X. Chen, H. Fang, T.-Y. Lin, R. Vedantam, S. Gupta, P. Doll´ar, and C. L. Zitnick, “Microsoft coco captions: Data collection and evaluation server,” arXiv preprint arXiv:1504.00325, 2015.

[8] 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, pp. 3128–3137.

[9] G. Kulkarni, V. Premraj, V. Ordonez, S. Dhar, S. Li, Y. Choi, A. C. Berg, and T. L. Berg, “Babytalk: Understanding and generating simple image descriptions,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 35, no. 12, pp. 2891–2903, 2013.