Convolutional Neural Network Exploration for Indoor Object Detection

1. Abstract

The painting trade is the last one to enter the construction process as their job involves painting the finished product of trades like carpentry. This creates a vulnerability to the painting cashflow as they are the last to be paid. By automatizing the painting process painting companies will lower their expenses. I present an exploration of using Convolutional Neural Networks (CNN) as a means to later create an AI capable of making an accurate cost estimate. This automatization will reduce manpower needed in smaller companies thereby alleviating some of the workload. This exploration achieved a total accuracy of 69% with live object detection.

2. Introduction

How can small painting companies increase their efficiency using AI? This exploration is a proof of concept for an Industrial-Painting robot. Before making a painting robot there first needs to be a visual detector for the robot to understand its environment. It was only natural to begin this project with an object detection algorithm. According to Greentechpainting there are over 30 classes needed to be recognized for the Ai to be useful. With this knowledge it is planned to make a robust AI capable of both indoor and out-door object detection of paintable surfaces and non-paintable surfaces. The goal is to apply the object detection AI to a machine capable of making invoices in real time as an addition to distinguishing between what to paint.

This proof of concept needed to demonstrate that real time object detection was possible inside an app/program and that it was not dependent on the internet or external libraries to operate properly while taking into account a high accuracy rate. The proof of concept was built using a basic CNN.

The model was built in four stages: 1. Initialization 2. Training 3. Prediction 4. Evaluation.

Before expanding on the process it is first important to understand the benefits that came included with the data-set [7]. Sorted data into three files, training, test and validation each one containing 2 sub-folders for images and label data. Where the distribution was 70%, 7% and 17% respectively. YOLOv5 labeling format allows to increase the quantity of the dataset. The approach was to use bounding boxes and the coordinates provided by the labeling format to crop the useful data inside the primary images to be later used for the training of the program.

Training the model was rather simple since only the basics were used. Determining the accuracy of the model was done by first providing the model with the training images and comparing its result with the training label, then using the validation data as a way to see if it acquired applicable knowledge. The validation accuracy took precedence over the training accuracy, high validation accuracy would suggest that the model has learned the general patterns of the different classes. The prediction stage and the evaluation were done simultaneously using the testing data to see the culmination of the exploration.

3. Background

The painting trade is the last trade to arrive at the job site. As such they are particularly vulnerable as they are the last trade to get paid. As such, the path to maximum economic stability is painting for home owners, historical repairs, and the like. As the construction part is already sorted out, allowing the painting company to have a secure cash flow. For these opportunities to arise extensive resources go into the making of invoices.

When it comes to management of invoices one finds a myriad of articles about AI and their effects in economical workflows [5]. Invoices are holders of crucial information for companies but it's also time consuming [6]. That said there is not much research on automating invoices from its root. The goal of this paper is to create an AI capable of indoor object detection to be transferable into making invoices. While not specific for this field there are papers that can be applicable to this purpose. Such as a paper titled "Indoor Segmentation and Support Inference from RGBD Images" [3] where it is proven that AI can infer a supporting region, this application is useful for determining areas. Although for this particular case it would need to also understand the applicable areas that can be displaced. It would not make sense for the invoice to exclude a necessary portion because it could not identify that the object blocking the area could be moved.

A possible endpoint is to have an AI that detects objects predicting two big classes, paintable and non paintable. Once selected, have it specify what type of class it is. Ex. Paintable-Drywall. Simultaneously making a map that encapsulates the areas pertaining to the detected objects. Then using another database select the prices according to the specifications and using the area provided by the AI calculate the cost of service. While the specifics might get complex this paper is focused exclusively on making the proof of concept.

GreenTechPainting, which administered financial assistance to this exploration, has expressed how over 50% of their administrative manpower goes to the creation, storage and upkeep of Invoices. Hence automating this process would allow them to focus on other urgent tasks. That in mind the success of building an AI capable of making accurate invoices using object detection and other algorithms has the potential to replace jobs. While not detrimental for small companies where manpower is limited, for employees of bigger companies their job might be brought to an end.

4. Dataset

For this project the Indoor Object Detection dataset was used[7] which allowed for a faster initialization stage. As it already had divided itself into 76% training data, 7% testing and 17% validation [7]. Each holding separate folders for the image and label data. The data was saved using a YOLO format meaning that a single image could contain multiple objects. This style is not compatible for a basic CNN, therefore we needed to crop each object from each image and assign the appropriate label and as a by-product it increased the data set to a total of 5384 images each with their respective labels. Using this extracted data the final step of the initialization stage began. To resize all the images. The images were not all the same dimensions which proved fatal for the training of the model. To correct this misstep all images were resized to 150x150.

There are 10 classifications. The samples of each class are predominantly of older designs. Some samples are even of messy houses which is good since there is a chance that the invoice AI will need to operate in messy environments.

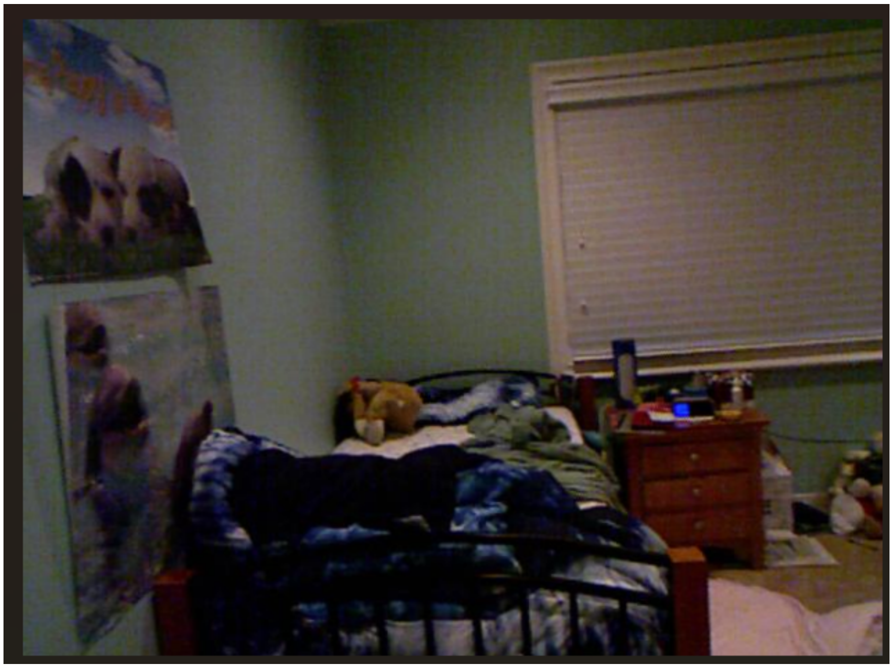

Fig. 1. This is an image before data extraction. From this image one can extract the table, bed and window. This is a good image for the model to extract the general pattern of a bed as not all beds will be nicely done.

5. Methodology/Models

Why use CNN? CNNs are designated to handle image Data. As such they are vastly used in object detection projects. Made of three features that simulate spatial awareness. Convolutional layers share equal weights per layer. Meaning that each layer is a convolution of the one before. Max pooling takes the maximum value in a predetermined array until the full image is processed and only the greatest values are preserved. Once both Convolutional and Max pooling layers are used in tandem the resulting matrix is flattened into a one dimensional array. This array is then processed by the third distinct feature, the Fully connected networks (Dense layer)[8].

How could the model work in real time with high accuracy? Due to the modest size of the dataset the model had to be trained with all available data. Hence the epoch size is preferred to be as low as possible in most cases. Where each epoch would be trained using all the images in the training folder. Then to test if the kernels had the appropriate patterns the validation folder was fed to the model. The first attempts had a proximate validation accuracy of 66%
Inspired by these initial results. Some experimentation on the architecture occurred. Unfortunately a particular phenomenon occurred. Any change to the architecture excluding weights,kernels and dropout would create stagnant results in the validation accuracy. It would be forever locked on 60.032%. This posed a great challenge throughout the rest of the model development stage. As the overfitting limited the convolutional layers to four with an equal number of max pooling layers. Increasing the epochs became a necessity to further elevate the desired accuracy.

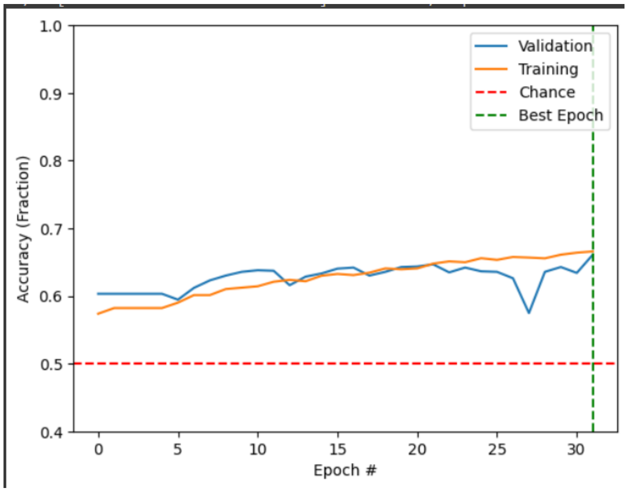

Fig. 2 . The first model shows promising results with a positive linear trend.

Observing the plot one question came to mind to increase the number of deep layers and their neurons help this particular model?

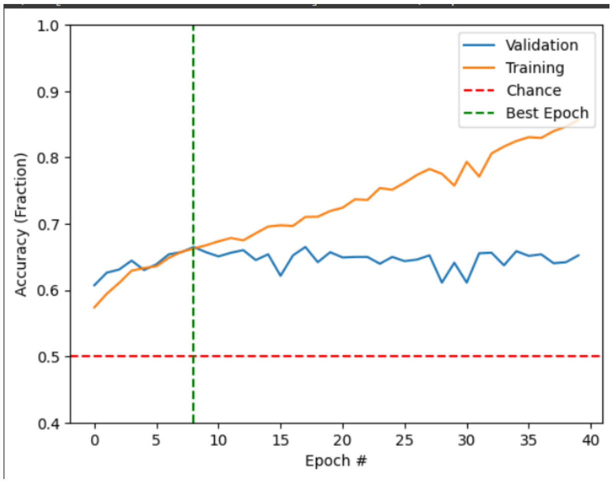

Fig. 3. The model experienced overfitting reaching a high 80% while the validation date fell behind. This result was somewhat expected as increasing the number of layers and neurons is known to cause this effect. Nevertheless with a limited number of convolutional layers this attempt was worth the try.

The following models were subsequently modeled to have about 2 deep layers and the focus increased in the dropout rate. Further experimentation proved two things: 1. There was a limit of 69% validation accuracy; 2. The validation accuracy was inversely proportional to the training data after the first epochs. After many days of diminishing results and a stagnation at 69.032% validation data it was time to move on.

With the best model saved (this allowed for the program to work without the internet) my attention shifted to developing a streamlit app where the model could perform predictions in real time. How could it do realtime predictions if it needed time to process the information? The result was quite simple after reflecting on the way the model was trained. The model took a 150x150 image and then made its prediction. With this in mind the streamlit app feeds the model an image every 5 frames thereby having real-time object detection.

6. Results and Discussion

The highest validation accuracy the model achieved was 69%. A few goals were accomplished. Real time object detection, no need for internet and proved to be a valuable experience. After discussing the results with my mentor it was determined that a larger dataset was crucial for obtaining higher validation data. As well as using a better algorithm one of the suggested options was YOLO. With this in mind the experimentation of synthetic data was encouraged. For the goal to use this AI to be of use the validation accuracy has to increase.

During the live object detection there was an interesting result. When presented with sofas or chairs it had a hard time distinguishing them from doors. This result did not happen when providing the testing images. Meaning that there was something going on with the real time data that was obscuring its judgment. This miss classification also occurred when presenting the model with windows that had trees behind it. Probably confusing the wood as a possible door it consistently predicted the windows as a door.

Further investigation into the nature of this problem revealed that the training images were mostly of older styles and not more modern ones. This is a precautionary tale for future models. There is a strong need for multiple styles inside each class. The model got good at recognizing older styles of sofas but had difficulty with other styles. Another observation to note is that when presented with trash bags or other items that it was not trained to predict it would consistently pick the class “Door”.

7. Conclusion

The exploration revealed that for this project to be successful there are three criteria that need to be met: 1. Large dataset, increasing the dimensions of the images will allow the filters to process more information as before with the image dataset worked on this project the addition of additional filters would result in them looking at a single pixel, aside from being minute it is also impractical. 2. Diverse dataset, containing varying degrees of angles,lighting and proximity. Doing so will allow for an increased understanding by the model of the particularity of each class. 3. Complex algorithm capable of predicting multiple objects inside the image. Possibly maximizing the efficiency of object detection as a byproduct. Further exploration into the field will provide efficient ways to combat the setbacks presented by lack of expertise. When training the model it would be beneficial to experiment with bigger batch sizes and epochs. Allowing the model to be trained for longer and with a variety of mini-batches.

The use of CNN can become an extremely computationally intensive process where the parameters and physical limitations of the hardware will limit the capacity of the project. [9].

8. Acknowledgements

I acknowledge support from Greentech Painting Inc. Guidance from Ronil Synghal who collaborated throughout the building of the object detection model. Special thanks to Chiang Mai, who provided the data set.

9. References

[1] Baviskar, D., Ahirrao, S., & Kotecha, K. (2020). *Multi-layout unstructured invoice documents dataset: A ... - IEEE xplore*. IEEE Xplore logo - Link to home. Retrieved from https://ieeexplore.ieee.org/document/9481217/?denied=

[2]W. Liu, D. Anguelov, C. Szegedy, D. Erhan, S. Reed, C. Fu, and A. C. Berg. SSD: Single shot multibox detector. In European Conference on Computer Vision (ECCV), 2016

[3] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus. Indoor segmentation and support inference from RGB-D images. In European Conference on Computer Vision (ECCV), 2012

[4]Tamilalagan, S. (2022). Invoice Generator Using Process Definition Document with Robotic Process Automation. *International Journal of Research in Engineering, Science and Management*, *5*(4), 178–180. Retrieved from https://journal.ijresm.com/index.php/ijresm/article/view/1989

[5]Arslan, H., Emre, Y., & Gormez, Y. (2023, August 25). *A deep learning-based solution for digitization of invoice images with automatic invoice generation and labelling - International Journal on Document Analysis And Recognition (IJDAR)*. SpringerLink. Retrieved from https://link.springer.com/article/10.1007/s10032-023-00449-4

[6]Lima, R., Paiva, S., Ribeiro, J. (2021). Artificial Intelligence Optimization Strategies for Invoice Management: A Preliminary Study. In: Sharma, H., Gupta, M.K., Tomar, G.S., Lipo, W. (eds) Communication and Intelligent Systems. Lecture Notes in Networks and Systems, vol 204. Springer, Singapore. Retrieved from https://doi.org/10.1007/978-981-16-1089-9_19

[7] Mai, C. (2022, July 21). *Indoor objects detection*. Kaggle. https://www.kaggle.com/datasets/thepbordin/indoor-object-detection

[8] Gibiansky, A. (2014). *Andrew Gibiansky :: math → [code]*. Convolutional Neural Networks - Andrew Gibiansky. https://andrew.gibiansky.com/blog/machine-learning/convolutional-neural-networks/

[9] Du, J. (2018, April 1). *IOPscience*. Journal of Physics: Conference Series. https://iopscience.iop.org/article/10.1088/1742-6596/1004/1/012029