An Efficient Floor Plan Classification with Optimized Image Features using Machine Learning

Karthik K
Artificial Intelligence Engineer
PropMix.io L.L.C.
Thiruvananthapuram, India
k.karthik@ieee.org

Safvan CK
Senior Artificial Intelligence Engineer
PropMix.io L.L.C.
Thiruvananthapuram, India
mail.safvan@gmail.com

Vinu Abraham Samuel
Artificial Intelligence Engineer
PropMix.io L.L.C.
Thiruvananthapuram, India
vinuabraham09@gmail.com

2022 IEEE 19th India Council International Conference (INDICON) | 978-1-6654-7350-7/22/$31.00 ©2022 IEEE | DOI: 10.1109/INDICON56171.2022.10039830

***Abstract*—Adding pictures is a necessary part of advertising a property for sale. Agents typically do not label images and even if they do they are not standards for such labeling. Floor plan is one of the essential categories of listing images that real estate portals would like to highlight and attract attention to. When volumes are small, manual annotation is not a problem, but there is a point where this becomes too burdensome and ultimately infeasible. Here, we propose an approach to radically increase the efficiency of such tasks. We present a novel algorithm to classify floor plan images of any type with the help of unique intrinsic image features such as mean saturation, dominant color extraction etc. and machine learning. The overall pipeline can accept any listing image, extract its features and predict whether the given image is a floor plan or not in a single shot. The overall experimentation shows that there is a performance improvement when comparing with a deep learning pipeline for the same and nearly 100% accuracy with the test data. We are also showing some additional experimentation results with other ML models apart from the best one.**

***Index Terms*—Floor Plan Classification; Real Estate Listing Images; Intrinsic Image Features; Classical Machine Learning; Low Cost Solution**

I. Introduction

There is a trend in the technical society that whenever there is a requirement to generalize a problem, people often use the easy method. But, is the easy method the accurate one? Or is the easy method the more efficient one? Deep Learning can be used to solve many complex tasks. But being powerful it has its own limitations in the solved scenario. Traditional ML techniques have advantages on this to reduce the complexity of the entire pipeline, it also helps to minimize the execution time and resource usage on a production environment. By efficiently tapping the features it also efficiently produces the output. In this paper the inputs that are fed to the algorithm are real estate listing images. Images that are scanned from other sources and normal images are also taken into account. The objective is to classify whether the given image is a floor plan or not by extracting the intrinsic features out of the image. This seems to be a simple problem statement while saying, but it requires an unerring feature extractor to make this happen. A deep learning binary classification could do the trick of classifying the images as either floor plan or not a floor plan. But we have used image processing algorithms to extract the above mentioned intrinsic features for the prediction. To do the extraction we have used a series of computer vision algorithms like mean saturation check, dominant color extraction, color palette extraction, number of lines, contour count extraction and variance of Laplace operation on an image which are clearly mentioned in section-III.

978-1-6654-7350-7/22/$31.00 ©2022 IEEE

II. Related Work

Floor plan drawings are an essential artifact of real estate - single/multi storied constructions. Systems were developed for efficient floor plan and rooms in floor plan classification using various technologies. This ranges from using manual identification of features to full automated feature detection using Deep Learning techniques. Most of these were focused on floor plan classification/building type classification based on design features of floor plan like rooms, halls etc. Even though we are focusing on whole floor plan classification/ classification of floor plan instead of rooms, halls, attics etc.. from other types of listings/design/architectural images, we have gone through papers which focused on recognition of various types of floor plan. [1,2,3] use low level features like text information, structural and semantic analysis for recognizing various types of floor plans. These systems mandate that the floor plans should be structurally and semantically different for efficient recognition. Other methods were focused on extracting features from images using SIFT and SURF techniques and classifying images based on these features. But these systems had the inherent problem of generalization. That is because these algorithms have different generalization methods as in rotation invariant, blur, warp transformation and scaling of the images. A comparative analysis shows that both SIFT and SURF algorithms are good for different kinds of images to perform well.

With development in deep learning techniques CNN's replaced most traditional feature extractors. [4] proposes CNN's for novel floor plan layout type classification. The method described in [5] uses transfer learning and CNN to classify floor plans. In this paper a pre-trained VGG net is used to obtain feature vectors of floor plans and these vectors are used for the classification process. Even though deep learning based classifiers excel in feature extraction and classification there are some problems in using them. 1) They need large amount of data to learn novel features from images. 2) Complex net work architectures are needed for learning complex patterns i images. 3) Requires more training time and resources. 4) Les control over the features that CNN's take for classification.

III. Proposed Solution

The entire floor plan prediction is divided into two parts feature extraction and prediction. The input image is pre processed and fed to the feature extractor block which is de tailed in the following parts . After the features are extracted, it is then passed to the prediction engine. Instead of convoluting the features and feeding it to fully connected layers, we are extracting some unique features from the images. With the help of those features we are doing the prediction in this pipeline. The unique feature that are extracted is as follows:

1) The mean saturation value of a floor plan image will be low when compared to a non floor plan image
2) The dominant color in a floor plan image is in a range tending to 255 .
3) The number of unique color palettes will be less in a floor plan image when compared.
4) The number of lines in the floor plan images will be higher when compared to a non floor plan image.
5) The Contours will be high in a floor plan image
6) The variance of a Laplace operation on the floor plan image will be high

Clubbing all these results together, the final prediction is obtained at higher accuracy and in low inference time. Let's look at the six algorithms in detail.

A. Mean Saturation Check

There are so many unnoticed features in an image that are so powerful and that can influence a decision in several ways. In the floor plan classification algorithm, there is such a feature which can influence the prediction. The saturation value in the HSV color space[6] has a significant impact here. The mean value of the saturation in a certain image can act as one of the features in floor plan classification. If the mean value is low, then it is likely to be a floor plan and if the value of the mean saturation is high we can say that the given image is not a floor plan. The value variations for a floor plan and a non floor plan image is shown in [Fig.1]. Experiments demonstrate that this can act as a critical feature in the algorithm. The algorithm for finding the mean saturation check is demonstrated in section III-A. Such low level features in an image is making significant impact in classical machine learning which is often left unexplored in many cases for different use cases.

■ Mean Saturation Check Algorithm
- Resizing into a common scale so that the features are standardized.
- Converting the image to HSV color space from the present color space.
- Take the saturation channel and find its mean value of the given image.

Fig. 1. Mean Saturation Values of a Non Floor Plan and Floor Plan image.

- Return the mean value.

Finding the mean can be mathematically represented as,

$$\hat{\mu} = \frac{1}{N}\sum_{i=1}^{N} x_i \tag{1}$$

where,
$\hat{\mu}$= Mean of the Saturation channel

N = Number of pixels in the channel

x_i= Pixel Value at nth position

B. Dominant Color Extraction

The color schematics inside a floor plan image and a non floor plan image have a distinct feature inside. When the floor plan image is processed, there is a common trend that is observed which is, the dominant color in a floor plan is tending to a value equal to 255 and the dominant color in the normal images is always less than this range.The dominant color range in an image can be calculated as ,

$$D(X) = max\sum_{i=0}^{m}\sum_{j=0}^{n} P(p_i, p_j) \tag{2}$$

where,
D(x)= dominant color range in the image

m,n = length and width of the image

p_i, p_j= pixel coordinates of ith row and jth column

P = color range of ith row and jth column

By taking this feature into consideration, the algorithm becomes more robust in determining whether the given image is a floor plan or not. The variations of the values for a floor plan image and non floor plan image is shown in [Fig.2]. The algorithm for dominant color extraction is mentioned in Section III-B.

■ Dominant Color Extraction Algorithm
- Resizing into a common scale so that the features are standardized

Fig. 2. Dominant Color Range of a Non Floor Plan and Floor Plan image

Fig. 3. Number of Color Palettes of a Non Floor Plan and Floor Plan image

- Find the color and the frequency of the color is appeared in the image
- Find the index of the maximum occurred color value
- Return the dominant color with the help of the index

C. Color Palette Extraction

A color palette, in the digital world, refers to the full rang of colors that can be displayed on a device screen or othe interface, or in some cases, a collection of colors and tools fo use in paint and illustration programs. The color palette reveal a lot about the electronic design of the device or technology and its visual capabilities for human users.[7] . We have foun out a uniqueness in the color palette of the floor plan image too which act as our feature for prediction. This is extracte from an image by computing the number of unique colors i the image which is mathematically represented as:

$$|D(X)| = \left| \sum_{i=0}^{m} \sum_{j=0}^{n} P(p_i, p_j) \right| \tag{3}$$

where,
|D(x)| = Number of unique color palettes

m,n = length and width of the image

p_i, p_j = pixel coordinates of i^{th} row and j^{th} column

In the general floor plan images the unique color palettes are less and vice-versa in non floor plan images. Just imagine the scenario, as the floor plan is build with least plain colors and geometry and non floor plan images are not. This can act as the other feature for determining whether the image is a floor plan or not.

- ■ Color Palette Extraction Algorithm
  - Resizing into a common scale so that the features are standardized
  - Find the color and the frequency of the color occurring in an image
  - Return the number of unique colors

D. Number of Lines in the Image

The number of lines in the images also have significant importance in the determination of floor plan. As mentioned in Section III-C, floor plan is made of straight lines and geometric shapes. This paved a way to extract some of the aspects of floor plan as the features for the prediction. An image showing the proof of concept that the lines actually matters is shown in [Fig.4]. The number of geometric lines in the floor plan

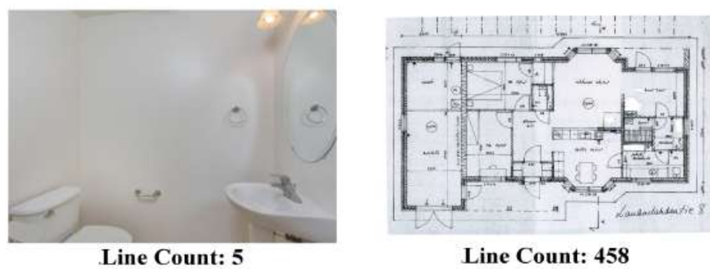

Fig. 4. Number of lines a Non Floor Plan and Floor Plan image

image will be high as it quantifies the statement mentioned in the above paragraph. In the normal images the lines will be less than in the usual scenario.

- ■ Line Counting Algorithm
  - Resizing into a common scale so that the features are standardized
  - Converting the image to grayscale for easy computation
  - Blurring the image a bit to remove the noise artifacts
  - Finding the edges so that the line detection will become easy
  - Detecting the line coordinates using Hough lines algorithm [8] with optimal parameters
  - Return the number of line coordinates

E. Contour Count Detection

When we are looking at a floor plan, we can primarily say that it has lines and geometric shapes arranged according to engineering mathematics in a plane back-ground. What we are not seeing primarily is its secondary aspects. We already said that it consist of different geometric shapes. Considering those shapes into account, consider the outlines that bind the shape to form something in the image. Such outlines are known as contours. When the contour count was experimented in the aspect of this, it is observed that the number of contours in

Fig. 5. Number of Contours of a Non Floor Plan and Floor Plan image

Fig. 6. Variance of laplace operation of a Non Floor Plan and Floor Plan image

the floor plan images are high and the number of contours in the non floor plan images are less. This led to the idea of using the contour counts as the feature for prediction in the ML model.

■ Contour Count Detection Algorithm

- Resizing into a common scale so that the features are standardized
- Converting the image to grayscale for easy computation
- Inverting the image to binary image
- Sharpening the image to smoothly tap the outlines from the image
- Finding the contours using find contours method [9].
- Return the number of contours

F. Process Laplacian

A Laplace filter is a filter used to detect the edges in an image. It is used to compute the second derivative of an image with respect to the measure at the rate the first derivative changes in the image. So this is used to determine the edges of an image. [10]. The variance of the Laplace operation can be calculated as:

$$var(X) = 2\sigma^2 \tag{4}$$

in this stage, experimentation are being conducted by applying a Laplace filter on the image. If the variance of the Laplace operation is high in the image, it is said that clearly distinguished edges can be seen. In account to that, it is observed that in floor plan images we are getting higher variance value when compared with a non floor plan image.

■ Process Laplacian Algorithm

- Resizing into a common scale so that the features are standardized
- Standardizing the image
- Finding the Laplace transformation on the image.
- Return the variance of Laplace transformation

IV. Dataset

The Dataset used in the experiment is a mixture of CubiCasa5K dataset[11] which contains 5000 samples annotated into over 80 floor plan categories. Also we have crawled public websites to take images for the non floor plan category. The non floor plan category includes the images of houses, buildings, rooms, indoor and outdoor scenes. The reason why we stuck on to these images was so as to make the entire environment aligned with real estate.

As a whole, for this experimentation we used 10000 images, 5000 floor plan images and 5000 non floor plan images . Out of the 10000 images, 7000 images were used for training the Machine Learning model and 3000 images for testing the model.

Fig. 7. Sample images from the dataset

V. Floor Plan Classification

The entire pipeline consists of two parts; feature extraction and classification part. An initial labeled sample is needed in order to train the model. We used well known supervised Ensemble Machine Learning method Random Forest algorithm for classification with these intuitive image features. From the input image, feature extraction is performed only after resizing into a common scale so that the features are standardized. For easy computation of number of lines, contours, etc. the image is converted into grayscale. After doing all the feature extraction described in the Section III, the labeled dataset is split for training and testing. Once trained, the algorithm is tested in the labeled test set and if the accuracy of the model is good enough it can be applied to an unlabeled samples.

The base of the Random Forest algorithm is the Decision Tree algorithm, in which the different branches are weighted with a certain probability. The algorithm chooses the attribute from the training dataset that best splits the subset by minimizing a certain function (typically an entropy or Gini index). As a result of this training process, branches are weighted, thus making it possible to apply the algorithm to an unlabeled image sample for its classification.

VI. Result Timing Analysis

The research is conducted on several instances. A VGG-16 Neural Network model was trained on this 1000 images and the results were noted. It is found that the results were having some discrepancies in the prediction of some images.

The Purpose of training a Neural Network model is to compare the results with our original experimentation. In that also we have explored some series of Machine Learning Models. The Result of the Models with the accuracy of the same is marked in [TABLE.1].

TABLE I
Model Experimented with its accuracy

Sl. No	**Model Table**		
	Model Used	***Accuracy(%)***	***Inference Time(s)***
1	VGG16	83%	3.8s
2	SVC	68%	2.4s
3	XGBoost	90%	2.1s
4	Voting Classifier	73%	4s
5	Random Forest	100%	2s

From the above table it is clear that random forest is the Machine Learning Model that is giving better accuracy and faster inference time. The confusion matrix of the test set predicted on RF Model and VGG-16 is shown in Fig.8 and Fig.9. The predicted label is on the x axis and the true label is on the y axis.

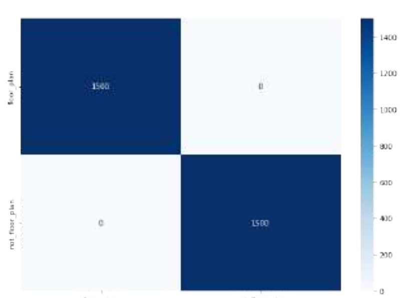

Fig. 8. RF Model on test dataset of 3000 images

A. Timing Analysis

The timing is an important factor in every machine learning production aspect. The graph which shows the time taken for the Inference in a Deep Learning model with the RF model is

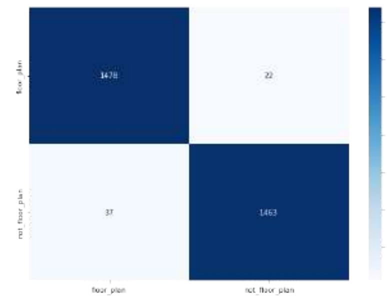

Fig. 9. Trained VGG-16 CNN Model on test dataset of 3000 images

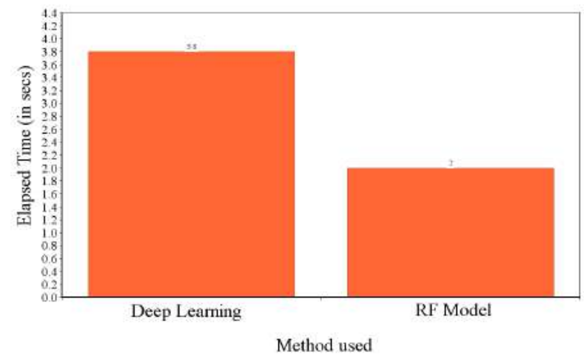

Fig. 10. VGG-16 vs RF Model in Time Analysis

shown in Fig.10. The elapsed time is in y-axis and the Method used is in x-axis.

All the experiments were done on an i7/16GB/6GB 16060ti machine.

VII. Conclusion

We have proposed a pipeline to identify whether a given property image is of a floor plan or not. Our method is better for its simplicity, speed and accuracy along with vision generalization capability in this domain. The experiments prove that the proposed algorithm performs comparatively much faster and is more robust than the deep learning models which produce the same result.

Acknowledgment

The authors would like to thank the organizations, 'Innovation Incubator Advisory Pvt. Ltd.' and 'PropMix.io L.L.C.' for the extended support to make this work possible.

References

[1] Sheraz Ahmed, Marcus Liwicki, Markus Weber, and Andreas Dengel. Improved automatic analysis of architectural floor plans. In 2011 International Conference on Document Analysis and Recognition, pages 864–869. IEEE, 2011.

[2] Sheraz Ahmed, Marcus Liwicki, Markus Weber, and Andreas Dengel. Automatic room detection and room labeling from architectural floor plans. In 2012 10th IAPR International Workshop on Document Analysis Systems, pages 339–343. IEEE, 2012.

[3] Sheraz Ahmed, Markus Weber, Marcus Liwicki, and Andreas Dengel. Text/graphics segmentation in architectural floor plans. In 2011 International Conference on DocumentAnalysis and Recognition, pages 734–738. IEEE, 2011.
[4] Yuki Takada,Naoto Inoue, Toshihiko Yamasaki, Kiyoharu Aizawa. "Similar floor plan retrieval featuring multi-task learning of layout type classification and room presence prediction". In 2018 IEEE International Conference on Consumer Electronics (ICCE).
[5] Xingyu Guo, Yang Peng. Floor Plan Classification Based on Transfer Learning. In 2018 IEEE 4th International Conference on Computer and Communications (ICCC)
[6] (2008) HSV Color Space. In: Li D. (eds) Encyclopedia of Microfluidics and Nanofluidics. Springer, Boston, MA. https://doi.org/10.1007/978-0-387-48998-8_656.
[7] https://www.techopedia.com/definition/6547/color-palette
[8] https://www.cse.unr.edu/ bebis/CS474/Handouts/HoughTransformPaper.pdf
[9] Suzuki, S., and Be, K. (1985). Topological structural analysis of digitized binary images by border following. Computer Vision, Graphics, and Image Processing 30, 32–46. doi:10.1016/0734-189X(85)90016-7.
[10] Yuanhao Gong, Wenming Tang, Lebin Zhou, Lantao Yu, Guoping Qiu. Qurater Laplacian Filter for Edge Aware Image Processing. In 2021 IEEE International Conference on Image Processing(ICIP).
[11] https://zenodo.org/record/2613548