AiLECS lab
Ai for law enforcement and community safety

Report TN22/05 (Honours thesis submission). Date of publication: March 21, 2022.

Geolocation of images taken indoors using convolutional neural network

Yukun Yuan

Monash University, Melbourne, Australia

Abstract

In this paper, I propose and explore a method for image location classification. Most existing works concentrate on outdoor scenes as scenery or an iconic landmark make it easier to point out the location. Few researchers have addressed the issue of indoor scenes. Although indoor images increase the difficulty of tracking geolocation, it is necessary to respond to this shortcoming as many crimes happen indoors. To address this problem, I propose a method for indoor image location classification by segmenting patterns of extracted objects from images. Specifically, I extract objects from images. Then, based on the accuracy levels of the bounding boxes of specific kinds of objects in the image, I only crop that kind of objects from original images. Moreover, I segment patterns from the extracted objects and crop those patterns by thresholding techniques. To classify images by these segmented patterns, I employ convolutional neural networks. Experimental results in the dataset of hotel rooms across the globe show promising accuracies, which witnesses that my method contributes to ultimately identifying the hotel chain which the image belongs to from the hotel dataset.

1. Introduction

The generally accepted use of the term "image geolocation" refers to the location where a non geo-tagged image is taken on a map [1]. Further, image geolocation is attracting widespread interest in the field of law enforcement [2]. For example, in the aspect of fighting against wildlife crimes, generation of a genetic database which contains geographical information allows molecular forensic tools to determine the source location of seized tiger parts [46]. In addition, image geolocation assists police to investigate people-trafficking crimes. Since sex traffickers regularly post photos of victims in their online advertisements [3], image geolocation is set to become a vital factor to find victims and prosecute perpetrators by taking advantage of these postings, which are clues for investigators to determine where the photos were taken [4]. Although indoor images increase the difficulty to track images for both computers and humans [47], it is necessary to respond to these crimes in order to intervene, prevent and prosecute to a large extent.

Fortunately, many approaches have been proposed to geolocate images. For instance, metadata with geographical identification has received much attention to recognize geolocation of data [9].However, is not always available because images can be uploaded without metadata and are not trustworthy because metadata supported by location-aware interfaces can be tampered easily [2]. An alternative way to geolocate images is the analysis of the image itself. For example, [7, 8, 9, 10] perform the task of geolocation only by similar scenes in the dataset of images, which illustrate that image analysis stands out for the ability to intelligently estimate geolocation of images only by the content of images. In addition to being based on the clues of similar senses in images, similar objects in images are more useful in the geolocation task for other images. For example, in order to identify the image's geographic location, [11] exploits landmarks appearing in images and matches images by these objects. Up to now, few researchers have addressed the issue of indoor scenes, since outdoor scenes can more easily be pointed out the location given scenery or an iconic landmark. However, indoor scenes also contain decorative patterns which may be discriminating enough to distinguish specific geolocations [12]. This argument is also supported by [13], where certain pieces of furniture are witnessed to be found more in certain regions of the world.

As for the aspect of extracting objects from images, there are three main kinds of frameworks. R-CNN generates region proposals to extract features in each candidate region and refine them to eliminate duplicate detections [16]. The second category [17, 18, 19] generates bounding boxes in the same way as the first kind of framework but only considers local information instead of global context when making predictions. Instead of using a kind of disjoint pipeline system illustrated in the two previous categories, YOLO [20] optimizes object detection in a single neural network, which end-to-end directly generates a faster and more accurate model.

To make the system more robust, annotation of high-level knowledge about different classes of the dataset is recommended [21]. For instance, giving a specific indoor object class its position in the images assists

indoor navigation [21]. Annotation of specific types of buildings, such as skyscraper, house and hangar in images can rebuild the dataset, which contained 70,000 terms from WordNet, to become the largest available dataset of scene categories with 899 categories and 130,519 images [22]. In experiments of Bashiri et al. [23], each image in this dataset can be represented by one chosen object extracted from the image.

In this paper, I propose a method for image classification where patterns of objects in images are fed into models. I first extract objects from images based on YOLOv3 which is a system to identify specific objects in images by generating and classifying bounding boxes based on each region of the image in a single neural network [20]. Through the comparison of accuracy levels of the bounding boxes of specific kinds of objects in the image, I was able to crop beds confidently. Then I used a commercially available package, thresholding, to segment patterns from extracted objects and put them into convolutional neural networks for image classification. Finally, experimental results offer compelling evidence for the contribution of my method to classification of 97000+ images from hotel chains across the globe.

The rest of the paper is organized as follows: Section 2 briefly reviews related studies. Section 3 illustrates the proposed method elaborately. Experimental results are presented in Section 4 and discussed in Section 5. Limitations and future work of this project are discussed in section 6.

2. Background

2.1 Image Geolocation

The term “image geolocation” has come to be used to infer the location at which a non geo-tagged picture has been acquired [1]. Recently, as sex traffickers regularly post photos of victims in their online advertisements [3], image geolocation is set to become a vital factor. For investigators attempting to find victims and prosecute perpetrators, these online advertisements can be used as evidence if investigators can determine where the photos were taken [4]. The US National Human Trafficking Hotline found hotels and motels to be a common venue for sex trafficking, with nearly 10 percent of known trafficking cases reported to them in 2016 taking place in a hotel or motel [5]. Thus, there has been considerable interest in finding the location of images taken indoors.

Although sometimes images can be automatically assigned with geographical information during capture [9], this is not reliable as it suffers from the situations where images are shared without metadata or do not carry trustworthy location tags as they are not resilient to tampering [2]. Fortunately, image geolocation with analysis of images themselves can compensate for this shortcoming [6].

2.2 Image Analysis

Image geolocation based on images themselves has received much attention. This has led a number of researchers, for instance [7, 8, 9, 10], matching similar scenes in images to investigate geolocations. Instead of similar scenes, quantities of images can only be compared by similar objects in images [11]. In the experiments of Zheng et al [11], landmarks were extracted from images and then used to cluster images. However, indoor images are neglected by recent literature, since there are no visible clues in outdoor images, including a well-known monument, such as the Eiffel tower, or natural scenery, such as tropical landscapes [12].

Fortunately, [12] underlines that indoor scenes also contain decorative patterns which are discriminating to distinguish specific geolocations. This argument can be proved by Liu et al., as they found that world regions show statistically significant variation in decorative element prevalence [13]. Thus, it is suspected that it is possible to geolocate indoor images by the objects included in the images.

2.3 Object Extraction

In general terms, object extraction can be defined as identifying specific objects in images by the determination of the presence or absence of specific features in image data [14]. Recently, the field of object extraction has made significant advances riding on the wave of convolutional neural network architectures [15]. There are three main frameworks:

To extract features in each candidate region in an image, at first, R-CNN and its variants generate potential bounding boxes based on each region. After that, these proposed boxes will be classified and then refined to eliminate duplicate detections. Since such a complex pipeline must be precisely tuned independently, the speed of this framework is limited [16].

Rather than refining bounding boxes, in DPM [17], OverFeat [18] and Deep MultiBox [19], each bounding box is along with a single score which is corresponding to the likelihood of containing any object. Although this kind of framework can perform single object extraction by a single class prediction, the bounding box only sees local information instead of global context when making a prediction.

Instead of using the region proposal or sliding window used in those two techniques mentioned above, YOLO simultaneously classifies all bounding boxes in a single neural network [20]. As the system can be optimized end-to-end directly, YOLO is an extremely fast and accurate model [20].

2.4 Annotated Dataset

An automated dataset is more reliable and efficient to classify. To be more specific, if high-level knowledge

about different classes of the dataset is accessible, the system can be more robust [21].

Negri points out that in the specific scenario of indoor geolocation, annotating datasets by identifying objects which are mostly associated with a certain geographical location give clues on the task of image geolocation [12]. This argument is supported by the experiments of Liu et al. [13]; they detected pieces of furniture such as art decorations, plants and books across six continents and then combined the analysis of residential ornamentation with the analysis of local geographical information. The results show that, by identifying decorative patterns and regional cultural decorative behaviour of furniture, images can be distinguished from different regions.

In addition, Afif et al. [21] proposed an annotated dataset by giving indoor object class its position in the images to assist indoor navigation. Xiao et al. [22] rebuilt the dataset, which contained 70,000 terms from WordNet, by annotating specific types of buildings, such as skyscraper, house and hangar in images and the final dataset reached 899 categories and 130,519 images, which provided the largest available dataset of scene categories. In experiments of Bashiri et al. [23], each image in this dataset was represented by one chosen object which was extracted from the image. After using the annotated dataset to train and test models, the classification performed well even when facing many challenging situations such as image rotation, intra-class variation and images variation. Negri also claims that annotating predominant objects in the images can boost the model's accuracy and make the architecture more robust when facing adversarial attacks [12].

As for our project, I will annotate the original data and then use the new dataset to train the model for indoor image geolocation.

3. Methodology

In this section, I first present the motivation and framework of the method of this project and then detail the generation of patterns. Lastly, I introduce how to use the extracted patterns to perform image location classification.

3.1 Motivation

While many works have achieved encouraging results on image geolocation, most existing works identify the image's geographic location by matching similar senses or objects in outdoor images, which ignores indoor images without scenery or an iconic landmark. However, due to the fact that victims of human trafficking are often photographed in hotel rooms [24], identifying these indoor scenes is undergoing a revolution in interest in these trafficking investigations. I am inspired by the fact that indoor scenes also contain decorative patterns which are discriminating to distinguish specific geolocations [12], as shown in Figures 1-4. The images in these four pictures are from four different countries (Dubai, Japan, Italy, Denmark), where certain pieces of furniture are witnessed to be more likely found in certain regions of the world [13].

I hypothesised that details of objects in a room image could give away the location of the image. To be more specific, after the photos are transformed into quantities of data points, features such as carpet, beds and paintings on the wall can be used to match against the database of images. This method can be applied for a wide range of hotels no matter whether they are independent or in large chains. In terms of hotels which are independent or part of very small chains, it may be possible to classify images by discriminating decorations.

On the other hand, for those hotels in larger chains, while the shared standards of interior decoration can look quite similar at first glance, the real value lies in getting the number of candidates to a small enough number that a human investigator could follow up on all of them. Thus, the aim of the task is to contribute to methods that can be ultimately used to identify the hotel chain which the image belongs to from the hotel dataset.

Figure 1. Example of hotel room in Dubai [42]

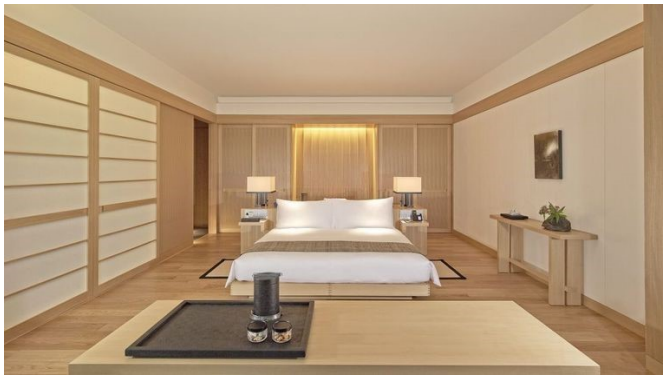

Figure 2. Example of hotel room in Japan [43]

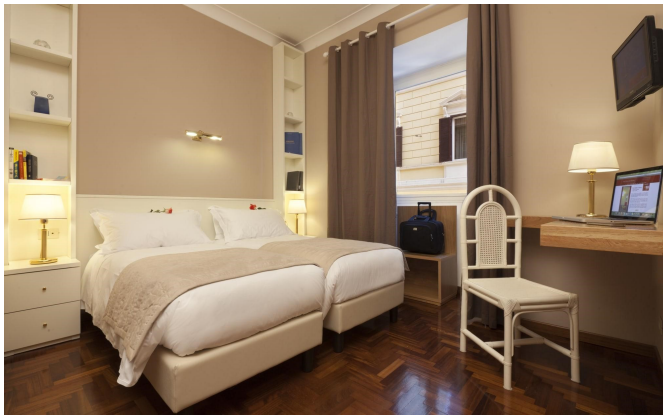

Figure 3. Example of hotel room in Italy [44]

Figure 4. Example of hotel room in Denmark [45]

3.2 Framework

The framework of our proposed method is divided into two parts. The first part is to produce final image representation by extracting pattern representation from objects in images, which is summarized in Figure 5. At first, objects are extracted by YOLO, which is a pre-trained convolutional neural network for object detection. After that, patterns are segmented from objects by thresholding techniques which is chosen by comparing results of different segmentation methods.

The second part is to put the segmented patterns into deep CNNs, e.g VGGNet and Xception, which are convolutional neural networks for image classification. In the context of image classification, a randomly initialised model is first compared with a pre-trained model. To quickly identify which method is more appropriate in my case, I conduct a small experiment by sampling a small subset of images and feeding them to both a scratch model and pre-trained model separately. Finally, I scale up the data by segmenting patterns from all images and use the chosen model to classify images.

3.3 Pattern Generation

Figure 5. During the process, beds are detected and cropped from the image and then patterns are segmented from beds.

3.3.1 Object Extraction

In order to investigate which hotel chain the image belongs to, I chose to extract objects from the image, as objects in images are witnessed to be useful to identify where the image is taken. For example, [11] exploits landmarks appearing in images and matches images by these landmarks. [25] uses just a few dozen objects like a house or graffiti on a wall which are extracted from images to represent the visual content of thousands of images. Meanwhile, indoor scenes also contain decorative patterns which are discriminating to distinguish specific geolocations [12]. For instance, in the paper [13], decorations of interior home spaces show statistically significant variation in different regions, as certain decorations of fourniture are witnessed to be found more in certain regions of the world.

To extract objects from images, YOLOv3 is used, which frames object detection as a regression problem by predicting spatially separated bounding boxes with associated class probabilities simultaneously in a single neural network [20]. There are two main reasons to illustrate that YOLOv3 is a reasonable choice. Firstly, by replacing all disparate parts of two previous kinds of frameworks with a single convolutional neural network, the system can be optimized end-to-end directly and thus YOLO is extremely fast [20]. In addition, this kind of framework uses features from the entire image to simultaneously predict all bounding boxes across all classes, which leads to an accurate model [20]. The steps to apply YOLOv3 for object extraction are as follows:

Step1: Download pre-trained YOLO to detect objects.

If the model is trained from scratch, it will take time to converge and the accuracy of the model could be low. While pre-trained model weights are used, it will show good accuracy and converge quickly for this task, as one of the target object classes (bed) is already represented in the pre-trained model. Thus, I load the pre-trained configuration and weights, as well as the class names of the COCO dataset on which the Darknet model was trained. The image size (416x416px), confidence threshold and the non-maximum suppression threshold applied in the code are also predefined values. Then, I run the basic function that will return detections for a specified image and the result is presented in Figure 6.

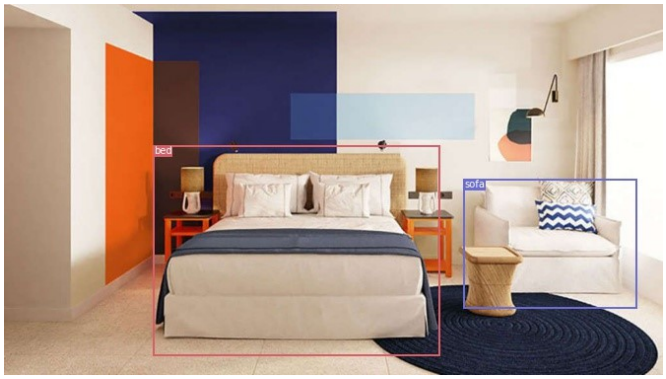

Figure 6. Detect objects in the image

Step2: Only crop beds from images, instead of all detected objects.

By exploiting the pre-trained YOLO classes, I am able to establish with high accuracy the bounding boxes of the beds in the image. This allows me to extract only the beds from those images that contain beds and index them for further processing. Figure 7 presents the result.

Figure 7. Only beds are cropped from the image and the results are save as a new .jpeg file

3.3.2 Pattern Segmentation

Step 1: Segment patterns from beds.

It is found that measuring the spatial patterns which emerge in residential decoration practices can reveal the presence of geographic culture hearths and/or globalization trends [13]. Furthermore, it is believed that features such as patterns in the carpeting, furniture, room accessories can be analyzed to narrow down the range of matched places [26]. Thus, I try to segment patterns from beds in this project. Generally, the pattern segmentation approaches can be categorized into two types:

1. Discontinuity detection based approach
   This is the approach in which an image is segmented into regions based on discontinuity. Due to intensity discontinuity, edges are formed and linked to form boundaries of regions [27].
   (a) Edge Detection
   Edge detection mainly works by detecting discontinuities in brightness to find the boundaries of objects within images [28].
   (b) Contour Detection
   Contour detection checks whether the continuous points have the same color intensity to determine the shape of closed objects [29].
2. Similarity detection based approach
   This is the approach in which an image is segmented into regions having a similar set of pixels [30].
   (a) K-Means
   K-Means clustering aims to partition N observations into K clusters with the nearest mean. A collection of data points are aggregated together due to certain similarities [31].
   (b) Color Detection
   Data points are detected and classified by using their RGB colorspace values [32].
   (c) Thresholding
   This is a non-linear operation which converts a gray-scale image into a binary image. The two assigned levels are whether below the specified threshold value or not. In other words, if the pixel value is greater than a threshold value, it is assigned one value (e.g. white), else it is assigned another value (e.g. black) [33].

Thresholding algorithms can be separated in two categories: a global thresholding technique which makes use of a single threshold value for the whole image and a local thresholding technique which makes use of unique threshold values for the partitioned subimages obtained from the whole image [34]. If the image background is relatively uniform, a global threshold value is more appropriate. However, if there is large variation in the background intensity, adaptive thresholding may produce better results [35]. From Figure 8 we can compare the results of these methods.

Compared with other segmentation methods, the thresholding method is more powerful to segment more precise target patterns from the bed.

Step 2: Crop segmented patterns from background.

After applying the threshold method to segment patterns, I applied morphology to clean extraneous spots and then get external contours. Then, I found the largest contour and drew the contour as white filled on a black background as a mask. The mask was then antialiased and put into the alpha channel of the input image. Finally, the cropped patterns departed from the background and were saved in png format. The result can be seen in Figure 9.

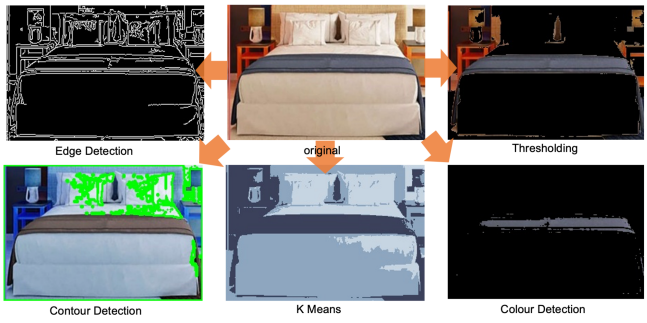

Figure 8. Compare segmentation results of different methods

Figure 9. The cropped patterns depart from the background

3.4 Implementation Details

In this section, I present the implementation details of experiments. Since almost all CNN architectures follow the same general design principles of successively applying convolutional layers to the input, periodically downsampling the spatial dimensions while increasing the number of feature maps, In this project, I select two state of the art convolutional neural networks, namely VGG16 and Xception.

3.4.1 VGG16

VGG16 is a convolution neural network (CNN) architecture that achieves 92.7% top-5 test accuracy in ImageNet which is a dataset of over 14 million images belonging to 1000 classes [36]. As detailed in Figure 10 [37], the 16 in VGG16 refers to it having 16 layers(convolution+ReLU and fully connected+ReLU) that have weights. Instead of having a large number of hyper-parameters, VGG16 is developed by only having convolution layers of 3x3 filter with one stride and always using the same padding and maxpool layer of 2x2 filter of stride. It follows this arrangement of convolution and max pool layers consistently throughout the whole architecture. In the end it has 2 fully connected layers followed by a softmax for output [36].

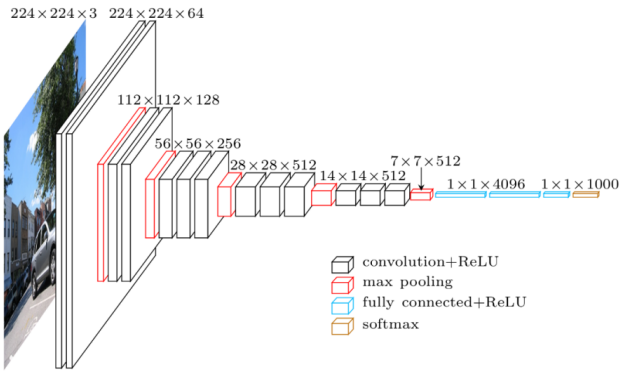

Figure 10. VGG-16 model architecture

3.4.2 Xception

Xception is a deep convolutional neural network architecture involving Depth Wise Separable Convolutions [38]. The architecture of Xception is shown in Figure 11 [38]. Based on an inception module, an optimal local sparse structure in a CNN can be approximated. That is to say, instead of being restricted to a single filter size, this image model block takes advantage of multiple types of filter size in a single image block, which then can be concatenated and passed onto the next layer [39]. A depth wise separable convolution can be considered as an inception module with a maximally large number of towers. The depthwise separable convolution is so named because it deals not just with the spatial dimensions, but with the depth dimension — the number of channels — as well [40].

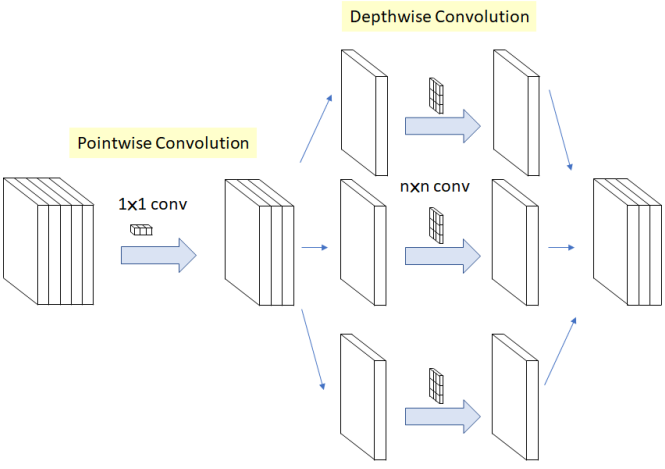

Figure 11. Xception model architecture

3.4.3 Random Initialisation (Scratch) vs Pre-trained Models

Training from scratch means to instantiate a model,not trained on any data and then train the model on the dataset. In contrast, instead of starting from training with randomly initialized weights, a pre-trained model can

use the weights from the previous network as the initial weight values. While a pre-trained model in a similar task often exhibits great results, two things can happen in transfer learning, namely positive transfer - at model pretrained on a big dataset performs very well when trained on a new dataset, and negative transfer - the model pretrained on a big dataset has a performance decline when trained on a new dataset [41].

To avoid the negative transfer, I conduct a small experiment, in which I make a small subset of the dataset of 10K images having the same class proportion as the original and train the model for both cases to quickly determine whether a scratch or pretrained model would be more successful in my case. As the records of accuracy compared in Figure 12 and Figure 13 show and that of loss compared in Figure 14 and Figure 15 the pre-trained model performs much more meaningfully than the scratch model, as the scratch one is overfitted caused by the limited number of data.

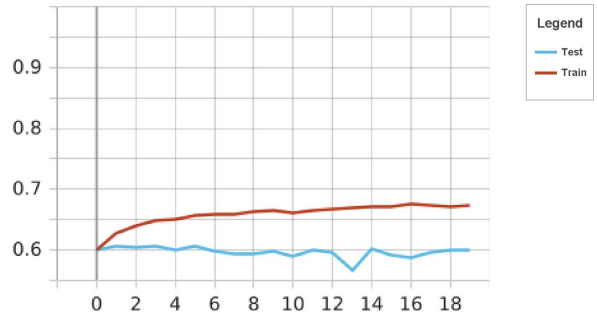

Figure 12. Accuracy in pre-trained model

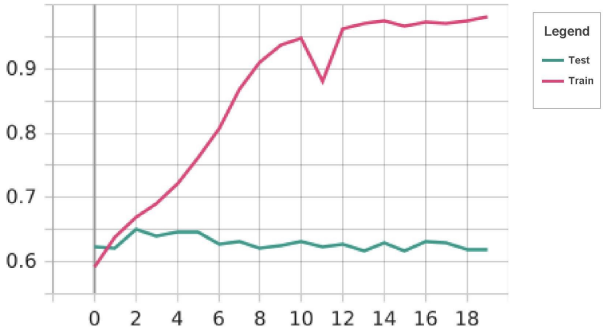

Figure 13. Accuracy in scratched model

4. Experiments and Results

In this project, I evaluate the method on the dataset obtained from a Kaggle competition [24]. The following describes the details of the experiments and results.

4.1 Date Collection

Hotel identification in general is a challenging fine-grained visual recognition task with a huge number of

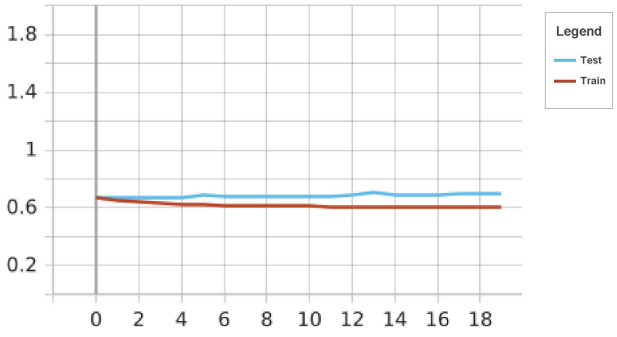

Figure 14. Loss in pre-trained model

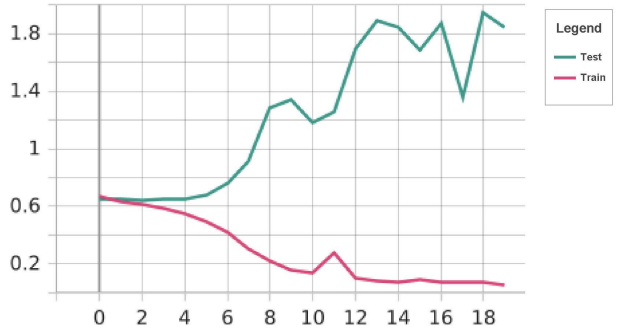

Figure 15. Loss in scratched model

classes and potentially high intra-class and low inter-class variation [24]. Fortunately, TraffickCam collected a rich dataset of photos of hotel room interiors which dramatically support research into this challenging task and create image search tools for human trafficking investigators. The dataset contains 97000+ images from around 7700 hotels from across the globe. In the parent folder containing the entire dataset, all of the images for each hotel chain are in a dedicated subfolder for that chain. Figure 16 presents an example in the dataset.

Furthermore, after the observation of hundreds of hotel images in the dataset, I find that the majority of images contain beds and decorative patterns on beds are useful to distinguish different hotel chains. Thus, I empirically decide to only focus on patterns in beds for model training.

4.2 Data Process

Since the methodology has only been applied on example images up to now, I tried to scale up data. During the process, I had to clean the data by removing corrupt image files. After more than 14 hours, the range of problem images narrows down and I find out that only when the image shown in Figure 17 was the input, there would be an error. Because there is only one image with error among the entire dataset, it is reasonable to throw out this specific image. After that, the code is successfully applied for the whole dataset. As a result, in each subfolder (one chain), the patterns of beds are cropped and

Figure 16. The original hotel image in the dataset

saved as new data.

I had to clean the data by removing corrupt image files, reorganise into appropriate directories for processing etc)

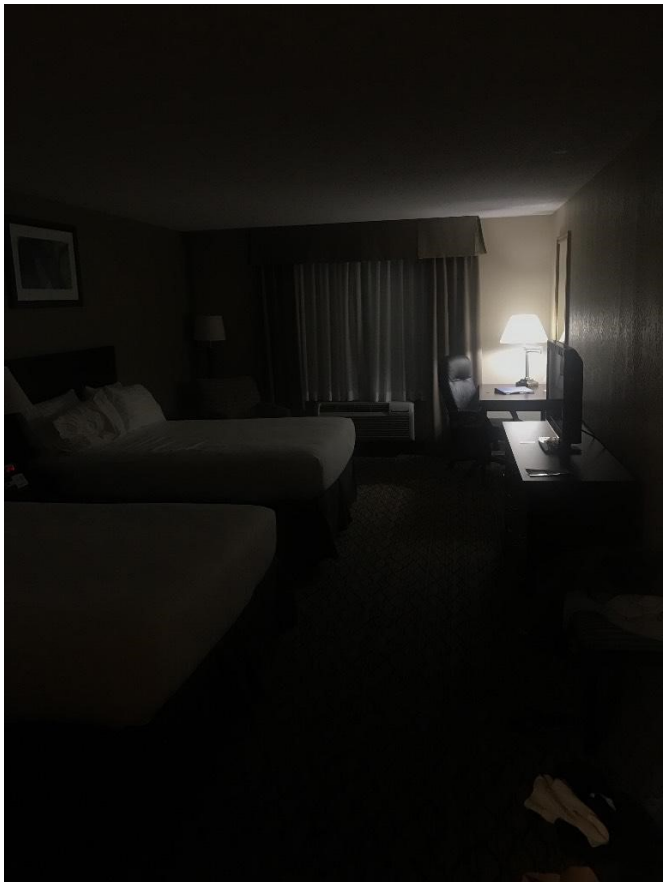

Figure 17. When I try to crop patterns of beds from this image, there is an error

In addition, there are two cares must be taken:

1. While there are 88 subfolders(hotel chain) in the dataset, only 24 subfolders contain more than 600 images and the details are shown in Table 1. With the target at making the model more confident, I only take these 24 subfolders into consideration.

Table 1. These 24 subfolders contain more than 600 images

No.	Hotel_chain_id	image_num
1	00	14,613
2	06	6,708
3	05	5,164
4	03	4,357
5	04	3,809
6	02	2,784
7	82	2,690
8	78	2,495
9	90	2,465
10	87	2,328
11	89	1,928
12	68	1,404

2. Some hotel rooms contain more than one bed. While more than one beds are able to be cropped from the kind of images like Figure 18, patterns of cropped beds are always the same. To avoid confusing the model with redundant data, I decide to only keep the patterns segmented from one bed from each image as input of the model.

Figure 18. The hotel room contains more than one bed

As the number of images is extremely large, it is inaccessible to manually pick one segmented pattern among two or three segmented patterns for each image. After comparing the quantity of pairs of patterns which are segmented from the same image containing more than one bed, I find that the segmented patterns of the first cropped beds are always with more discriminating patterns. Hence, I empirically decide to only keep the first segment patterns and delete other segment patterns for the same image in batches.

4.3 Evaluation on different tasks

4.3.1 Classify images in 25 classes

To keep balance among different classes, I randomly sample a similar number of images from each hotel chain. That is to say, for those subfolders with more than 500 images, only 500 images are selected as input of VGG16. Unfortunately, the average test accuracy is only 0.18.

After checking images of segmented patterns, I believe that images in which kept patterns are not related to beds should be considered as noise and removed. Here are findings about segmented patterns: Figures 19 – 22 list categories of cropped patterns from beds:

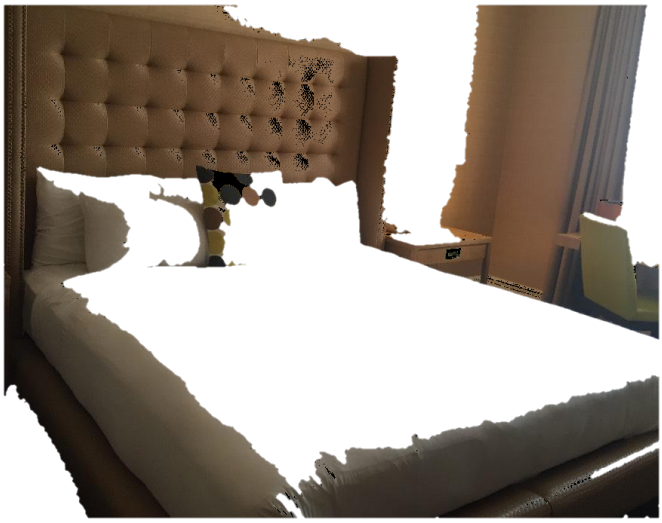

Figure 19. Pattern on headboard top rail

Figure 20. Pattern on blanket or throw

Figures 23 – 27 show categories of patterns of objects expect for bed:

4.3.2 Classify images without noise in 25 classes

After images of noise are removed, there is still no improvement and even lower in accuracy which is 0.16. Then, I suspect that high accuracy is not achievable just by the current limited number of data. Thus, I consider conducting an easier classification task.

Figure 21. Pattern on bed skirt

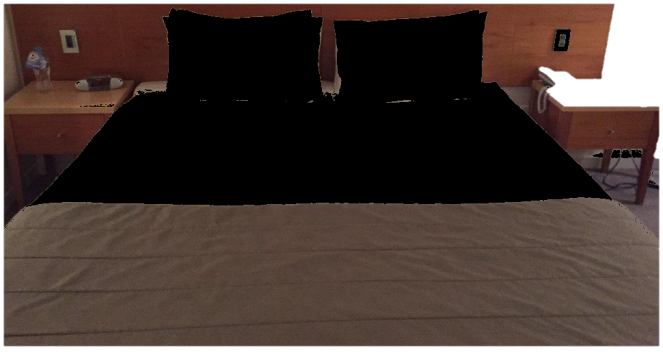

Figure 22. Pattern on duvet

Figure 23. Only pattern on carpet is cropped

4.3.3 Classify images in two classes

In the dataset, the subfolder named zero (0) indicates that the hotel image is either not part of a chain or the chain is not known, while the chains of images in the other 24 subfolders are identified. Thus, I merge these 25 subfolders into 2 subfolders, one is 00 and the other is a combined subfolder where all other 24 subfolders are included. Even though the test accuracy increases dra-

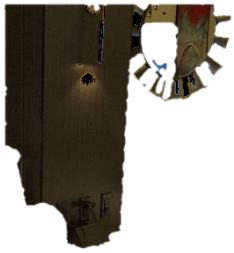

Figure 24. Only wall is cropped

Figure 25. Only sofa beside bed is cropped

Figure 26. Only table beside bed is cropped

matically to 0.76, the result is not too convincing as the distribution of images in these two classes are skewed where the size of 00 folder is two times smaller than that of the other subfolder.

Figure 27. Only individual items on bed are cropped

4.3.4 Classify images in two classes with balanced distribution

In order to make the distribution of labels approximately equal, thirty percent of images are randomly selected from the _0 subfolder and then the accuracy is 0.62. Even though the accuracy drops a little, records of both accuracy and loss shown in Figures 28 – 31 still illustrate that this methodology contributes to the task.

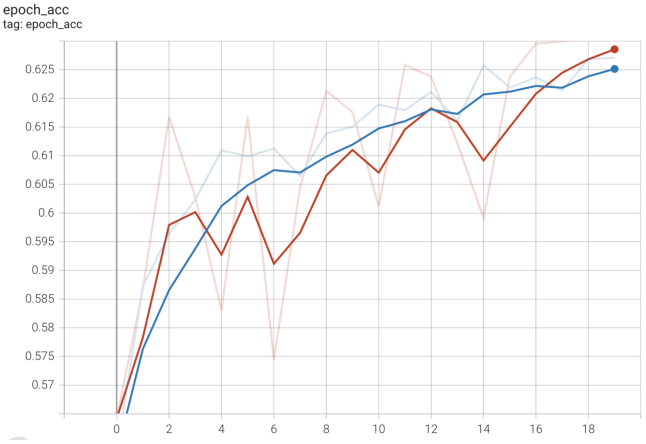

Figure 28. Records of accuracy in training and validation

4.4 Comparison with the state-of-the-art methods

In order to further increase the robustness of the methodology, I conducted the last task on two convolutional neural networks and compared their performance. As shown in Figures 32 – 35, there is no big difference when different CNNs are applied.

5. Discussion

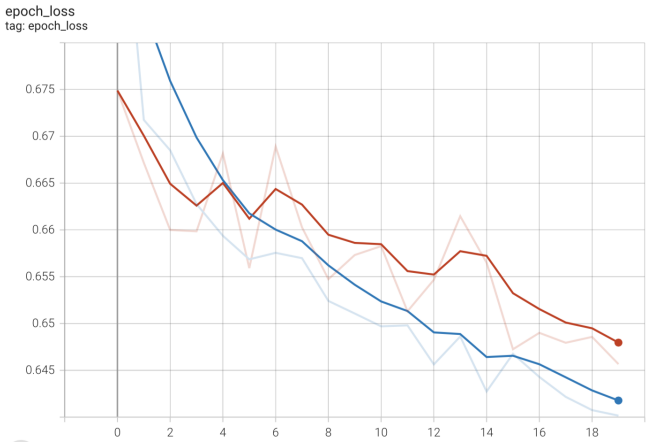

Figure 29. Records of loss in training and validation

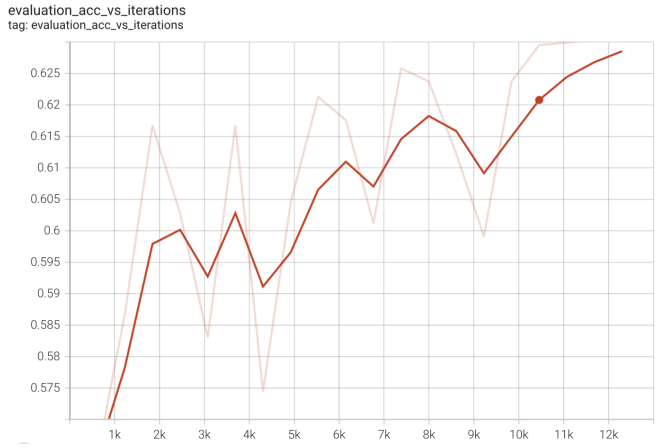

Figure 30. Records of accuracy in test

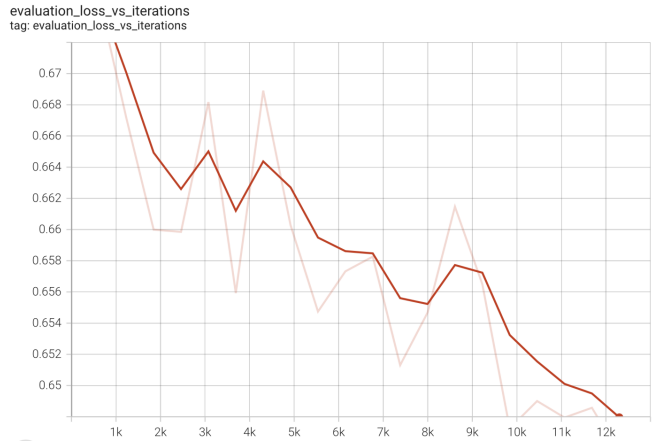

Figure 31. Records of loss in test

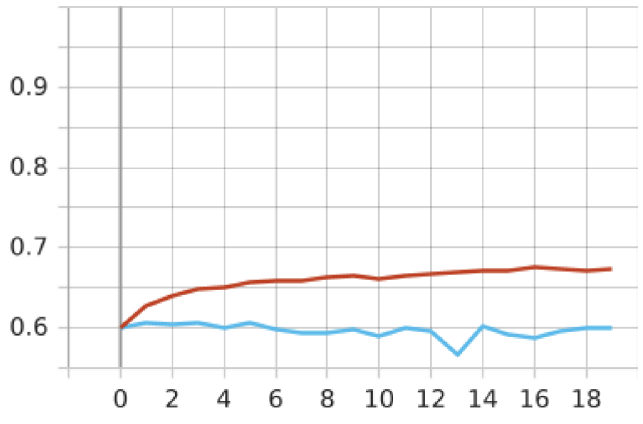

Figure 32. Records of accuracy in Xception

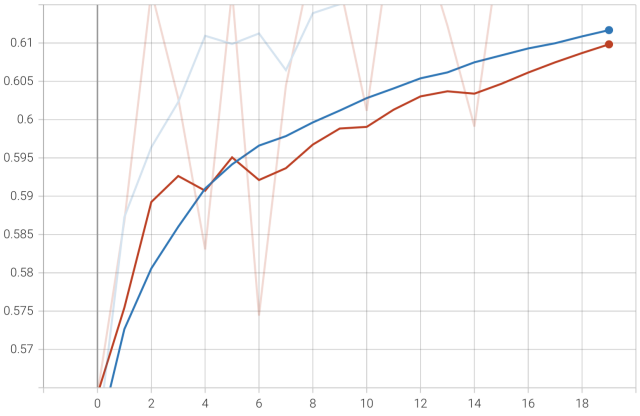

Figure 33. Records of accuracy in VGG16

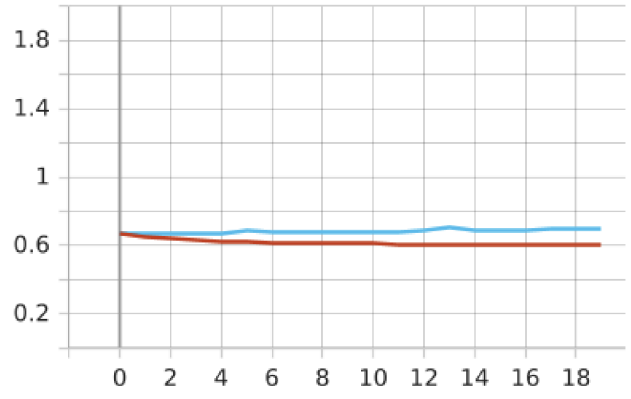

Figure 34. Records of loss in Xception

5.1 Objects correlated with hotel chain branding.

While there exist some discriminative features in the images from hotels, this dataset highlights one of the main challenges in the task to distinguish different hotel chains. On one hand, intra-class variation is high as not rooms within the same hotel chain have the objects with similar features. On the other hand, inter-class variation is low when recognizing a specific hotel chain from quantities of hotel chains when many kinds of objects are with similar features. With the aim at taking both intra-class variation and inter-class variation into account, the recognition of objects which affect mostly to distinguish different geolocation from images is essential.

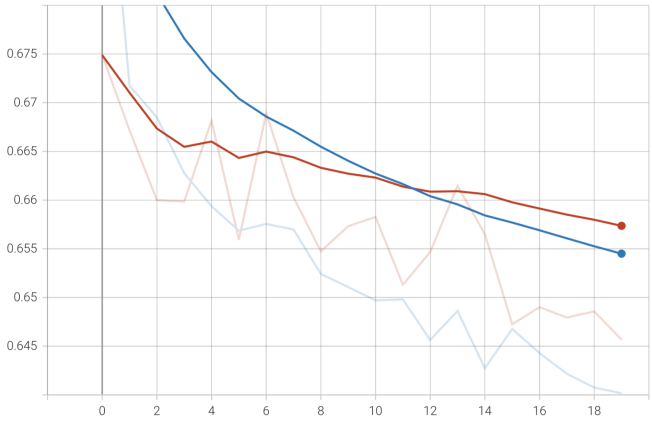

Figure 35. Records of loss in VGG16

Informed by the work of Negri,which sought to determine which kinds of objects are the most important in scenes and thus the models should focus on[12], I choose to select the kind of object empirically. By exploiting the pre-trained YOLO classes, I am able to establish with high accuracy the bounding boxes of the beds in the image. Thus, only beds are extracted from images and their patterns are put into models for training. The results in the No.4 experiment witnesses that decorative patterns on beds are discriminating to distinguish hotel chain brandings.

5.2 Object extraction technique.

In terms of object detection for recognizing patterns such as edges (vertical/horizontal), shapes, colours, and textures, Convolutional Neural Network (CNN) is commonly used recently [34]. CNN achieves excellent object detection accuracy by using a deep ConvNet to classify object proposals. In this task, I choose YOLO in which a convolutional neural network is involved, to extract objects, since it provides an end-to-end manner which is expected to lead to a much faster and more accurate model. After the step to crop extracted objects and save as new files, I check the results in each subfolder. Fortunately, I find that beds are cropped precisely from the original images. Furthermore, in terms of images with more than one bed, all beds are extracted successfully, which approves that YOLO is a powerful technique for object detection.

5.3 Reflection on performance.

As just mentioned above, I use YOLO to extract beds inside images and then segment patterns from beds. After that, I feed the segmented patterns to VGG16 for multiple parallel classifications. The tasks of the No.1 and No.2 experiments are to classify 25 hotel chains and the accuracies of both experiments are quite low. To improve the accuracy, the noise is removed in the No.2 experiment, while the accuracy still does not increase and even drop slightly. The difference between these two experiments shows that a smaller number of input results in worse performance of the model, as those removed noise decrease the quantity of input. In addition, the reason why those data are considered as noise is that the segmented patterns are not correlated to beds but to walls, carpets, sofas or desks, which illustrates that not only beds contain distinguishing patterns but other kinds of objects also contain discriminating patterns which are important to tell apart from different hotel chain brandings.

To be more specific, after the contribution of other kinds of objects are removed, the lower accuracy makes sense. Considering that the task to classify 25 hotel chains with high accuracy is inaccessible based on the limited number of data, I try to merge these 25 clases into 2 classes in the No.3 experiment, which not only enlarges the dataset but also eases the task. Expectedly, the accuracy increases dramatically. In addition, this result in No.4 experiment has further strengthened my confidence that patterns of beds can distinguish different hotel chains, as one of subsets is cut down two thirds of data to keep balance between classes. Regarding these two classes, one class indicates that the hotel image is either not part of a chain or the chain is not known and the other class presents that the chains of images in the subfolder are identified.

These results offer crucial evidence for a large amount of data. What is more, results provide additional support for differences between dependent or small hotel chains and large hotel chains. While the current work is generally sufficient to produce good results, I tried to obtain better performance by applying other models. As forecast, with the comparison of VGG16 and Xception, there is no significant difference. To some extent, the results of comparison eliminates the influence of different CNNs, as Convolutional Neural Networks are state of the art models for image classification.

6. Limitations and Future Work

It is plausible that a number of limitations could have influenced the results obtained. Firstly, the skewed dataset reveals the difficulty of collecting enough images in a wide range of hotel chains for classification and the disappointing results in the early experiments, which only have 500 images in each class, are evidence of the importance of enough data. When the number of data achieves the level to build a robust model, the task to find out which hotel chain the image belongs to will be more achievable. Another limitation is related to both time and hardware, during each experiment, it takes more than 14 hours to extract patterns from beds in the entire dataset.

What is worse, when there is an error in a few images, it is really time consuming to find out which images are with problems among the whole dataset. During training models, each epoch needs 40 minutes and thus in each

experiment only 20 epochs are conducted which is totally not enough to get the best performance. Furthermore, a major source of uncertainty in the project is that only beds are extracted and fed into models for classification. As mentioned in the previous part, the accuracy drops after the patterns on walls, carpets, desks or sofas are removed. This work is a proof of concept of using object detection and thresholding, but for best results this would be extended to include multiple objects.

Thus, further experimental investigations are needed to extract other kinds of objects and even combine many kinds of objects for image classification. In terms of pattern segmentation, unfortunately, I am unable to investigate the relationships between colours of objects and hotel chain brandings due to the fact that technology is hard to recognize the same colour in different conditions of lights. Future studies should examine whether there are more appropriate features except patterns of objects. In addition, parameters such as the confidence threshold and the non-maximum suppression threshold applied in the code during the process of object extraction are all predefined values and further exploration on different values of parameter are therefore required.

Acknowledgement

This report was originally submitted as a thesis paper for the degree of Master of Data Science (Honours) at Monash University. The supervisors were Associate Professor Campbell Wilson (Director AiLECS Lab), and Dr Gregory Rolan (Research Fellow, AiLECS Lab).

7. References

[1] Cristani, M., Perina, A., Castellani, U., and Murino, V. (2008). Content visualization and management of geo-located image databases. In CHI'08 extended abstracts on Human factors in computing systems (pp. 2823-2828).

[2] Wong, C. W., Hajj-Ahmad, A., and Wu, M. (2018, April). Invisible geo-location signature in a single image. In 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 1987-1991). IEEE.

[3] Itpro.co.uk. (2021). Retrieved 1 November 2021, from https://www.itpro.co.uk/strategy/26803/help-fight-sex-trafficking-by-taking-photos-of-hotel-rooms.

[4] Mitchell, K. J., Wolak, J., and Finkelhor, D. (2005). Police posing as juveniles online to catch sex offenders: Is it working?. Sexual Abuse: A Journal of Research and Treatment, 17(3), 241-267.

[5] Hotel/Motel Based Commercial Sex. National Human Trafficking Hotline. (2021). Retrieved 1 November 2021, from https://humantraffickinghotline.org/what-human-trafficking/sex-trafficking/hotelmotel-based-commercial-sex.

[6] "Finder (IARPA Research Program)," [Online]. Available: https://www.iarpa.gov/index.php/research-programs/finder, Accessed November 2016.

[7] Cristani, M., Perina, A., Castellani, U., and Murino, V. (2008, June). Geo-located image analysis using latent representations. In 2008 IEEE Conference on Computer Vision and Pattern Recognition (pp. 1-8). IEEE.

[8] Hays, J., and Efros, A. A. (2008, June). Im2gps: estimating geographic information from a single image. In 2008 ieee conference on computer vision and pattern recognition (pp. 1-8). IEEE.

[9] Zhang, W., and Kosecka, J. (2006, June). Image based localization in urban environments. In the Third international symposium on 3D data processing, visualization, and transmission (3DPVT'06) (pp. 33-40). IEEE.

[10] Snavely, N., Seitz, S. M., and Szeliski, R. (2006). Photo tourism: exploring photo collections in 3D. In ACM siggraph 2006 papers (pp. 835-846).

[11] Zheng, Y. T., Zhao, M., Song, Y., Adam, H., Buddemeier, U., Bissacco, A., ... and Neven, H. (2009, June). Tour the world: building a web-scale landmark recognition engine. In 2009 IEEE Conference on Computer Vision and Pattern Recognition (pp. 1085-1092). IEEE. [12] Virginia, N. 2020. A Hierarchical Geolocation of Indoor Scenes with Visual and Text Explanations using Deep Learning.

[13] Liu, X., Andris, C., Huang, Z., and Rahimi, S. (2019). Inside 50,000 living rooms: an assessment of global residential ornamentation using transfer learning. EPJ Data Science, 8(1), 4. [14] Radovic, M., Adarkwa, O., and Wang, Q. (2017). Object recognition in aerial images using convolutional neural networks. Journal of Imaging, 3(2), 21.

[15] Shrivastava, A., Gupta, A., and Girshick, R. (2016). Training region-based object detectors with online hard example mining. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 761-769).

[16] Uijlings, J. R., Van De Sande, K. E., Gevers, T., and Smeulders, A. W. (2013). Selective search for object recognition. International journal of computer vision, 104(2), 154-171.

[17] Felzenszwalb, P. F., Girshick, R. B., McAllester, D., and Ramanan, D. (2009). Object detection with discriminatively trained part-based models. IEEE transactions on pattern analysis and machine intelligence, 32(9), 1627-1645.

[18] Sermanet, P., Eigen, D., Zhang, X., Mathieu, M., Fergus, R., and LeCun, Y. (2013). Overfeat: Integrated recognition, localization and detection using convolutional networks. arXiv preprint arXiv:1312.6229.

[19] Erhan, D., Szegedy, C., Toshev, A., and Anguelov, D. (2014). Scalable object detection using deep neural networks. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 2147-2154).

[20] Redmon, J., Divvala, S., Girshick, R., and Farhadi, A. (2016). You only look once: Unified, real-time object detection. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 779-788).

[21] Afif, M., Pissaloux, R. A. Y. S. E., and Atri, M. (2019). A novel dataset for intelligent indoor object detection systems. Artificial Intelligence Advances, 1(1), 52-58.

[22] Xiao, J., Hays, J., Ehinger, K. A., Oliva, A., and Torralba, A. (2010, June). Sun database: Large-scale scene recognition from abbey to zoo. In 2010 IEEE computer society conference on computer vision and pattern recognition (pp. 3485-3492). IEEE.

[23] Bashiri, F. S., LaRose, E., Peissig, P., and Tafti, A. P. (2018). MCIndoor20000: A fully-labeled image dataset to advance indoor objects detection. Data in brief, 17, 71-75.

[24] Hotel-ID to Combat Human Trafficking 2021 - FGVC8 | Kaggle. (2021). Retrieved 28 October 2021, from https://www.kaggle.com/c/hotel-id-2021-fgvc8/overview

[25] Avrithis, Y., Kalantidis, Y., Tolias, G., and Spyrou, E. (2010, October). Retrieving landmark and non-landmark images from community photo collections. In Proceedings of the 18th ACM international conference on Multimedia (pp. 153-162).

[26] TechCrunch is now a part of Verizon Media. (2021). Retrieved 28

October 2021, from https://techcrunch.com/2016/06/25/traffickcam/

[27] Kumar, R., Arthanari, M., and Sivakumar, M. (2011). Image segmentation using discontinuity-based approach. Int. J. Multimedia Image Process, 1, 72-78.

[28] How to Perform Edge Detection in Python using OpenCV - Python Code. (2021). Retrieved 28 October 2021, from https://www.thepythoncode.com/article/canny-edge-detection-opencv-python

[29] How to Detect Contours in Images using OpenCV in Python - Python Code. (2021). Retrieved 28 October 2021, from https://www.thepythoncode.com/article/contour-detection-opencv-python

[30] Dik, A., Jebari, K., Bouroumi, A., and Ettouhami, A. (2014). Similarity-based approach for outlier detection. arXiv preprint arXiv:1411.6850.

[31] How to Use K-Means Clustering for Image Segmentation using OpenCV in Python - Python Code. (2021). Retrieved 28 October 2021, from https://www.thepythoncode.com/article/kmeans-for-image-segmentation-opencv-python

[32] Rosebrock, A. (2021). OpenCV and Python Color Detection - PyImageSearch. Retrieved 28 October 2021, from https://www.pyimagesearch.com/2014/08/04/opencv-python-color-detection/

[33] OpenCV 3 Image Thresholding and Segmentation - 2020. (2021). Retrieved 28 October 2021, from https://www.bogotobogo.com/python/OpenCV_Python/python_opencv3_Image_Global_Thresholding_Adaptive_Thresholding_Otsus_Binarization_Segmentations.php

[34] Rogowska, J. (2000). Overview and fundamentals of medical image segmentation. Handbook of medical imaging, processing and analysis, 69-85.

[35] Thresholding — skimage v0.18.0 docs. (2021). Retrieved 2 November 2021, from https://scikit-image.org/docs/stable/auto_examples/applications/plot_thresholding.html

[36] Step by step VGG16 implementation in Keras for beginners. (2021). Retrieved 28 October 2021, from https://towardsdatascience.com/step-by-step-vgg16-implementation-in-keras- for-beginners-a833c686ae6c [37] Nash, W., Drummond, T., and Birbilis, N. (2018). A review of deep learning in the study of materials degradation. npj Materials Degradation, 2(1), 1-12.

[38] Chollet, F. (2017). Xception: Deep learning with depth wise separable convolutions. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 1251-1258).

[39] Papers with Code - Inception Module Explained. (2021). Retrieved 28 October 2021, from https://paperswithcode.com/method/inception-module

[40] A Basic Introduction to Separable Convolutions. (2021). Retrieved 28 October 2021, from https://towardsdatascience.com/a-basic-introduction-to-separable-convolutions-b99ec3102728

[41] Wang, Z., Dai, Z., Póczos, B., and Carbonell, J. (2019). Characterizing and avoiding negative transfer. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 11293-11302).

[42] Home | The Meydan Hotel. (2021). Retrieved 3 November 2021, from https://themeydanhotel.com/

[43] Aman Tokyo, Japan 5 Star Luxury Hotel. (2021). Retrieved 3 November 2021, from https://www.luxurytravelmagazine.com/property/aman-tokyo-japan-5-star-luxury-hotel

[44] (2021). Retrieved 3 November 2021, from https://www.trivago.com.au/rome-44337/hotel/italia-1234191

[45] Magazine, W. (2021). Hotel Danmark — Copenhagen, Denmark. Retrieved 3 November 2021, from https://www.wallpaper.com/travel/denmark/copenhagen/hotels/hotel-danmark

[46] Karmacharya, D., Sherchan, A. M., Dulal, S., Manandhar, P., Manandhar, S., Joshi, J., ... and Hughes, J. (2018). Species, sex and geo-location identification of seized tiger (Panthera tigris tigris) parts in Nepal—A molecular forensic approach. PloS one, 13(8), e0201639.

[47] Thompson, W. B., Valiquette, C. M., Bennet, B. H., and Sutherland, K. T. (1996). Geometric reasoning for map-based localization. Computer Science Technical Report UUCS-96-006, University of Utah, Salt Lake City, UT.