Stage 6: CNN Classification Stage

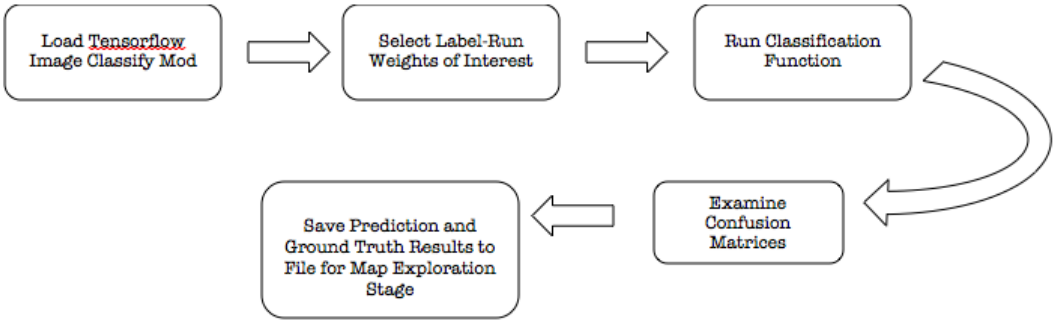


Fig 6.1 *– Classification Stage Workflow Overview*

- A script for selecting and loading model weights for batch test set classification runs and for assessing/validating train runs and inspecting misclassified images and training precision results.

In this section we outline the results and outcomes of the Image Classification Task as a whole. In reality the findings would emerge at various points during and over various iterations of the Training, Classification and Mapping workflow Stages while we developed further confidence and understanding of the data limitations faced, the machine learning processes involved and the living city which we were modeling.

6.2 Classification Benchmark Results

Inception v3 and MobileNet runs were carried out on the commonly used Flowers Dataset and also on the US Cities Research Case Study, to establish acceptable levels of accuracy. As a further reference point we also used weights obtained from the US Cities Data for classification runs on the LDD data(to determine if the US study would generalize onto the London dataset). The resultant metrics tables can be referred to in Figure 5.24.

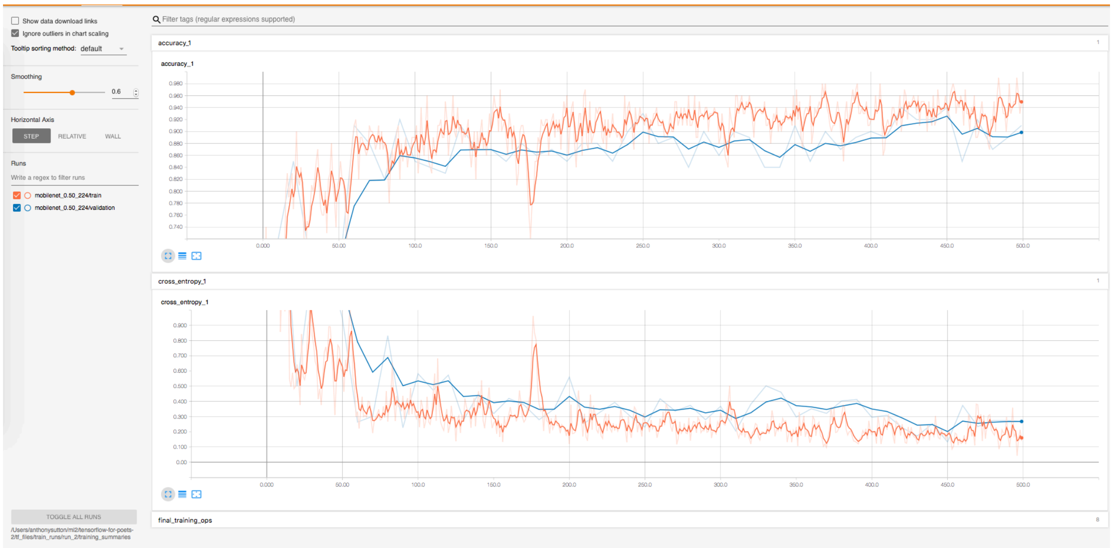


***Fig 6.2** – Flowers Benchmark Accuracy and Loss Curves*

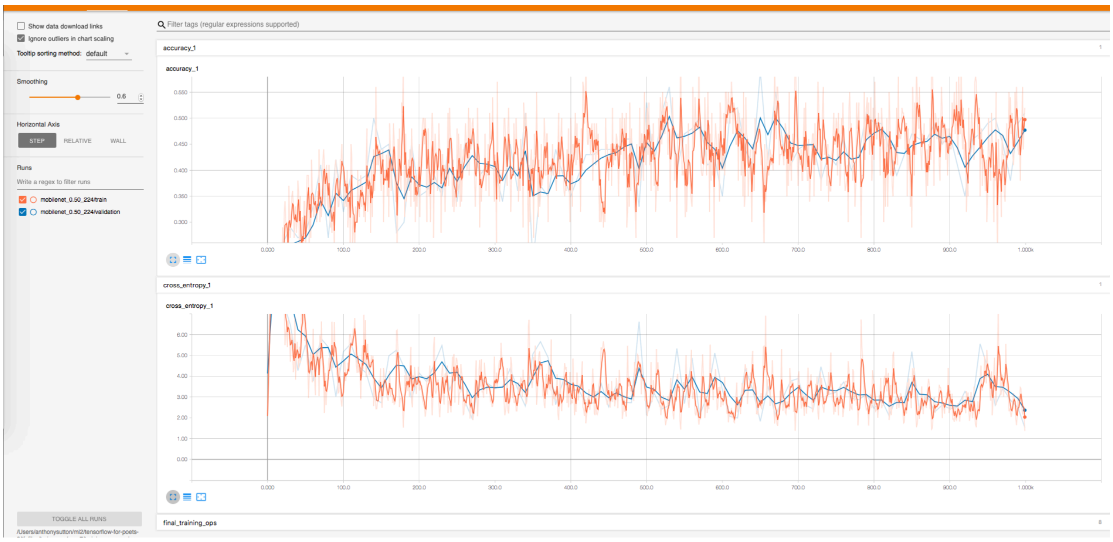


***Fig 6.3** – Accuracy and Loss Curves*
For Kang et al US Building Instance DataSet Retrained on MobileNet v2

6.2 Early Confusion Matrices

Best Fit AB to OSM Label Set

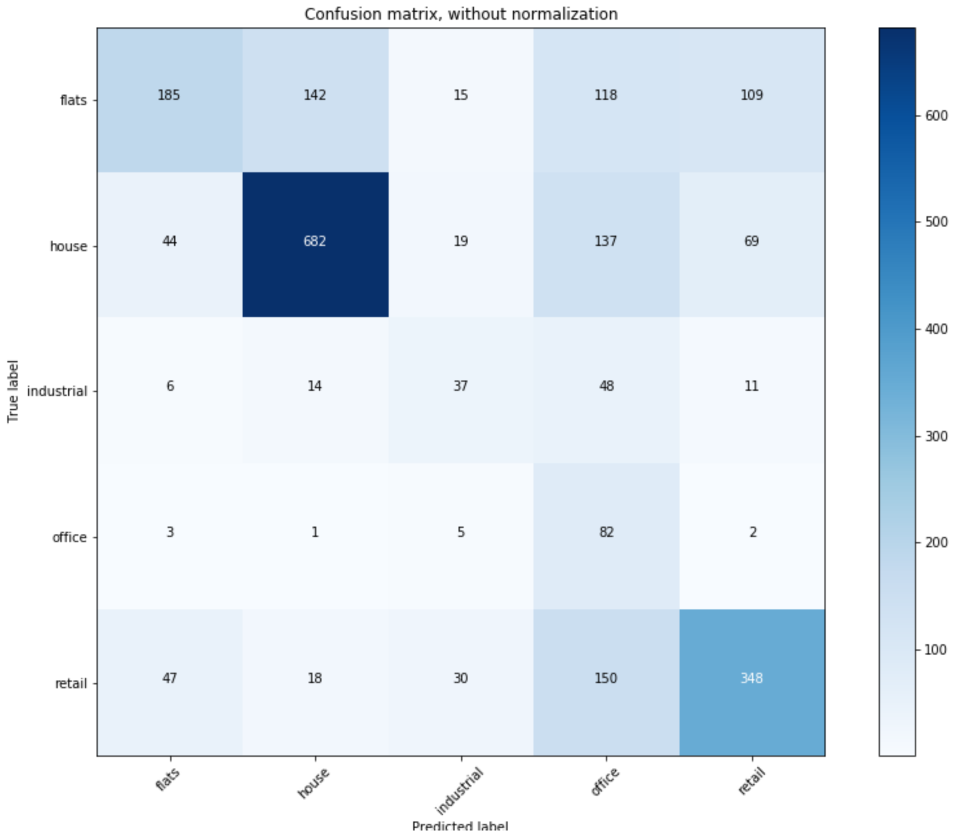
In Initial Training Runs of the Workflow we achieved accuracy rates of > 66 % on our best of class label run. This involved all 5 property types: Flats/Apartment, House, Retail, Industrial, Office. Greater accuracy was achieved by omitting offices or retail. The main problem types were the abundance of terraced properties found on many high streets and which might combine office, flat and retail. By applying the controls (limiting flats to those over 4 units) and by swapping out office or retail according to the mix of labels, we achieved reasonable precision accuracies from which to build a more in depth classification task.

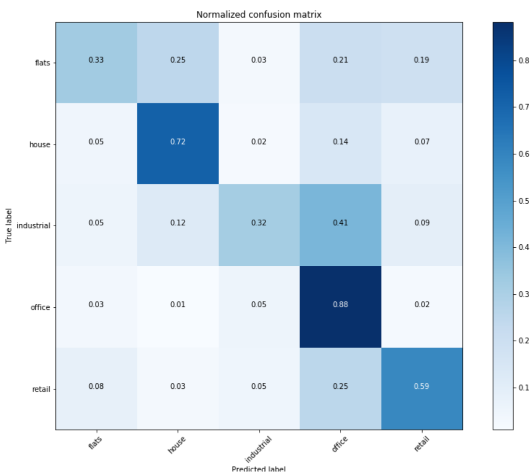
We also deployed label runs on one vs. all combinations, which achieved higher accuracy rates, but which due to their binary nature held greater expectations in terms of accuracy due to lesser number of categories.

We display two sets of Confusion matrix below(actual numbers predicted and with normalization). The former was for a Prediction run on Croydon, Brent and Tower Hamlets. The latter was for the whole of the London Wide Test Set (approx. 30000 records).

In the first example below, of the 951 Houses confirmed as true, 682 flats were correctly identified. Of the 589 flats, 185 were identified correctly but 142 were identified as houses(despite our efforts to limit to larger scale flat artifice types). This might reflect flat conversions into house object types. Of the 116 Industrial properties identified 37 were correct, however 48 were identified as office. This might reflect the similar non residential and business orientated visual and aesthetic nature of the two types. 348 retail were correctly identified, reflecting the role signage might have been able to identify a retail property type. 82 office were correctly identified, reflecting the filtering we applied to only focus on large scale office developments or Office blocks.

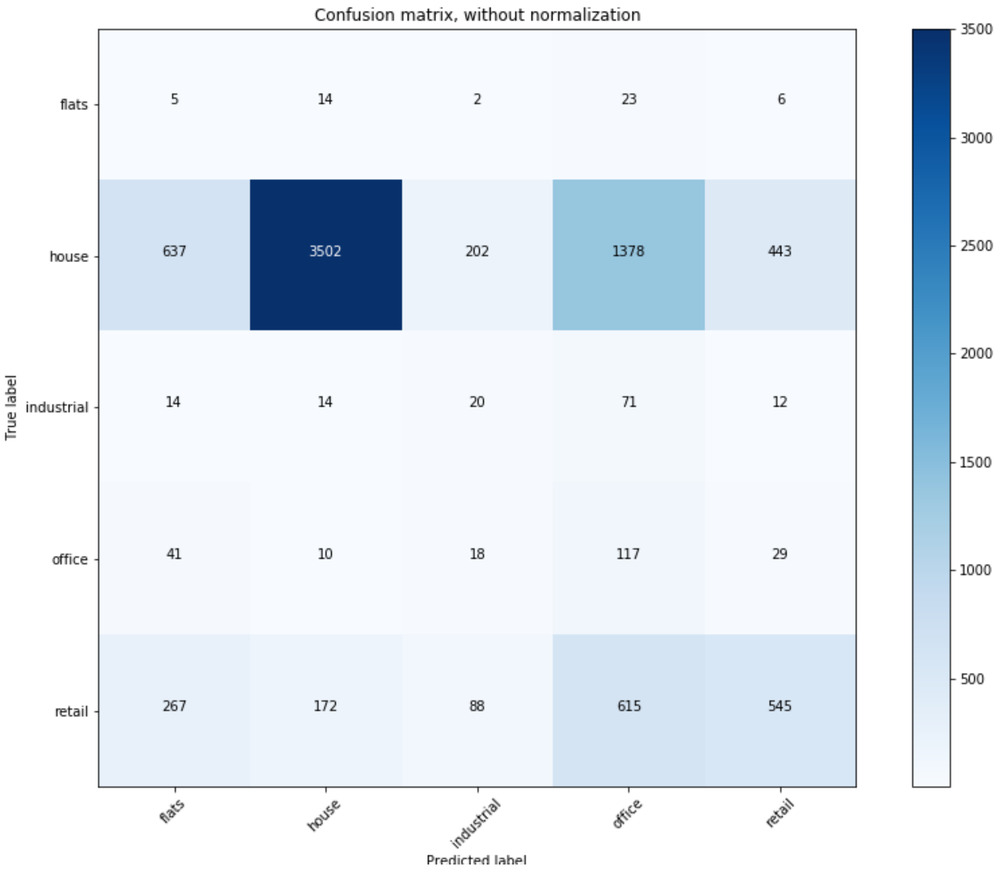
Similar patterns were found on the London wide dataset and can be explored using the Folium Map gallery in the following chapter.

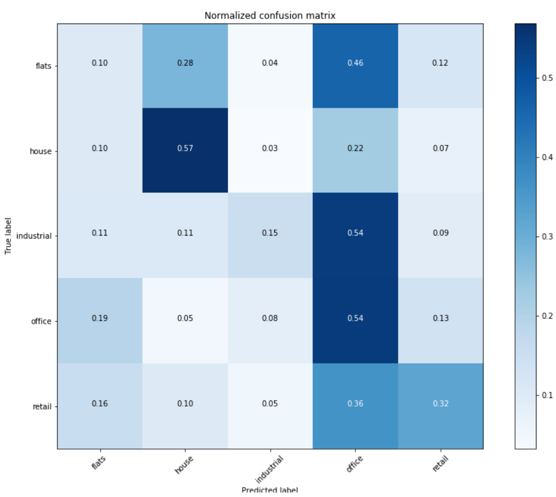




Following the Diagonal, the Dark Blue patch illustrates a good score, 682 of the House Class types were correctly identified. Normalized this equates to 72% correct labels for that class(see left matrix).

***Fig 6.5** – Croydon, Brent, Tower Hamlets Label Run*





Following the Diagonal, the Dark Blue patch illustrates a good score, 3502 of the House Class types were correctly identified. Normalized this equates to 57% correct labels for that class(see left matrix).

***Fig 6.6** – London Wide Run*

***Fig 6.7** – Best Fit label (One vs All) Accuracy and Loss Curves For House vs the Rest Class Labels*

6.3 Model Overfitting, Method Redesign and Problem Resolution Approaches

6.3.1 Model Overfitting

On deeper inspection of the Accuracy and Loss curves, further acquaintance with the Machine Learning literature and when attempting to recreate the initial borough and London wide selection Runs on validation runs we found significant signs of model overfitting.

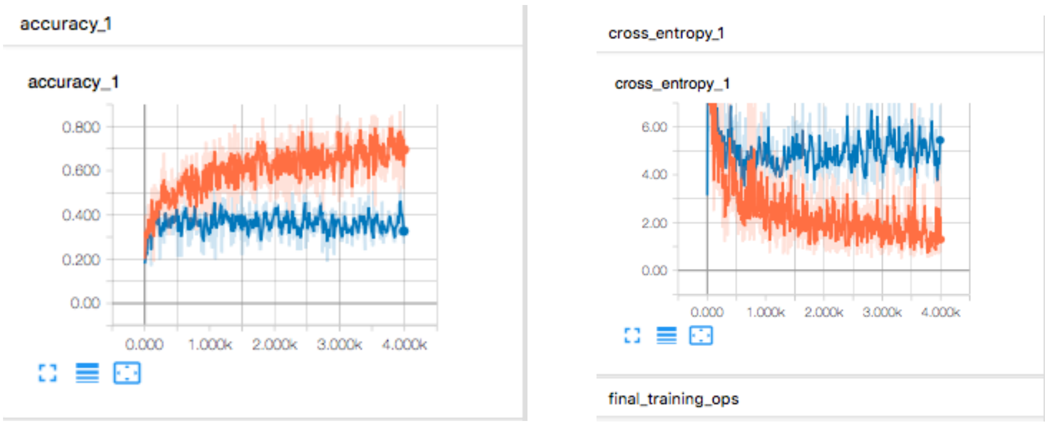


Fig 6.7 *– Best Fit Labels for AB to OSM (House, Flat, Retail, Industrial) and Overfitting*

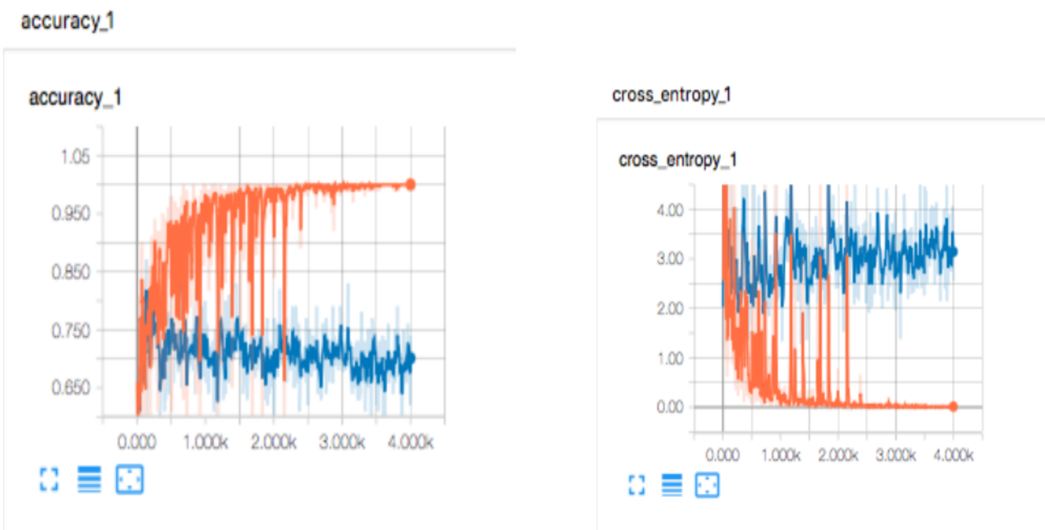


Fig 6.8 *– Best Fit Labels Overfitting - One vs the Rest(Misfit) Class*

In the examples above and below the precision and entropy curves display a wide gap between our training and validation results(the latter consistently lower and both frequently reaching peak loss or accuracy rates early on) and which show classic signs of model overfit.

Learning rate was also reduced in latter attempts but the acute variation seen in the orange training curve suggest that the low numbers of training records, short batch sizing and the continued presence of outlier visual images despite rigorous pre processing, type filtering and address/image match validation steps for the given label being trained were causing the model to overfit on the training data. This was confirmed when faced with new unseen data from the validation and test batches for the London wide Classification runs.

The relatively low test batches of the CNN Script would not be disposed to pick this up. The situation was confirmed when archived running label runs through the classification script and inspection of the confusion tables that were being created. The map tool, described further in the next section, proved useful in verifying address to street view image matching, as would the thumb gallery in providing a speedy lookup tool of the AddressBase and Planning Development description data combined with a ready actual visualization of the image being trained for manual verification.

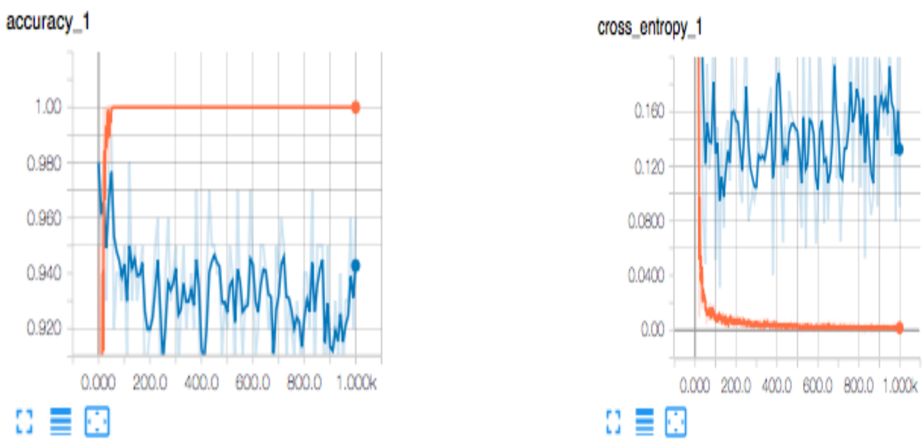
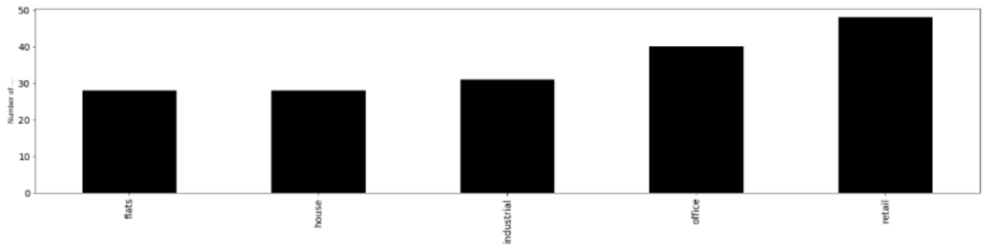


***Fig 6.9** – Best Fit Labels Overfitting - Hand Labelled New Build Large Scale Luxury Flat Development*

Quick Inspection of Results

Ground Truth - Property Type Distribution

In [44]:
```
#Check Distribution of Property Types FOR GROUND TRUTH
plotdat(df_ldd_label_join,'C_Truth')
```



CNN Prediction - Property Type Distribution

In [45]:
```
#Check Distribution of Property Types FOR CLASSFIER
plotdat(df_ldd_label_join,'Predicted_Label')
```

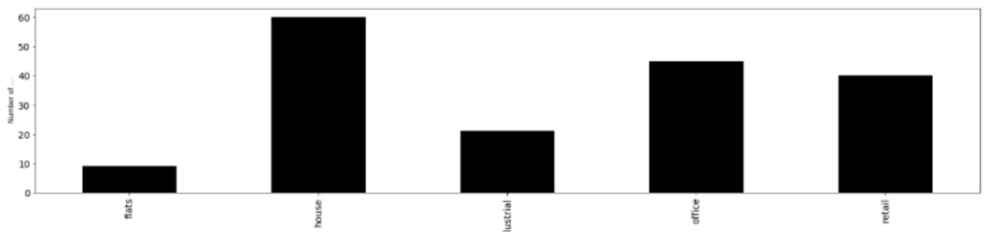


Fig 6.10 *– Typical Ground Truth vs Prediction Spread on Subsequent Classfication Runs. Splitting the Full Model into Test and Train earlier on in the Study would have identified this issue sooner.*

```
conf_mat = confusion_matrix(Ynp, Xnp)
acc = np.sum(conf_mat.diagonal()) / np.sum(conf_mat)
print('Overall accuracy: {} %'.format(acc*100))
```

```
Overall accuracy: 50.28571428571429 %
```

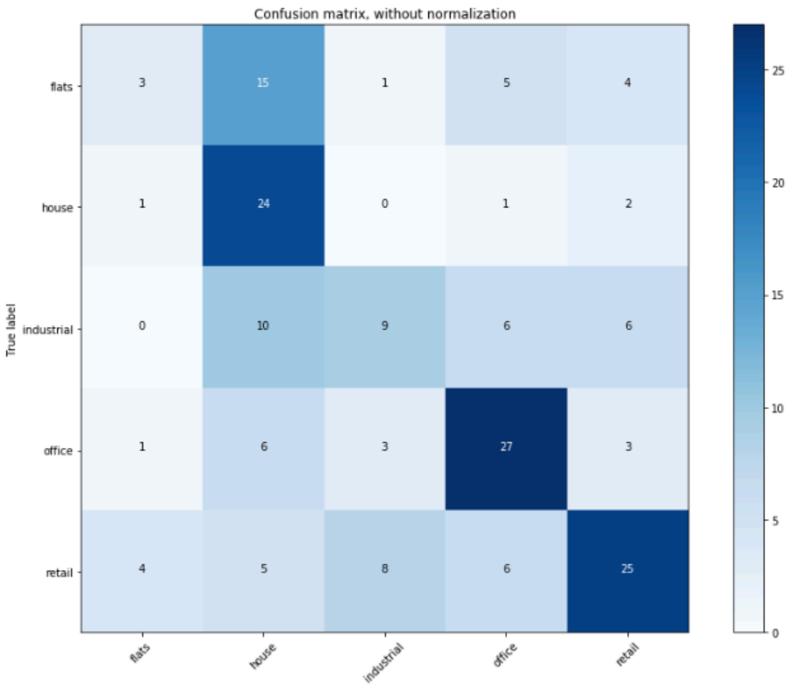


Fig 6.11 *– Confusion Matrix for Best Fit Labels demonstrating Overfitting*

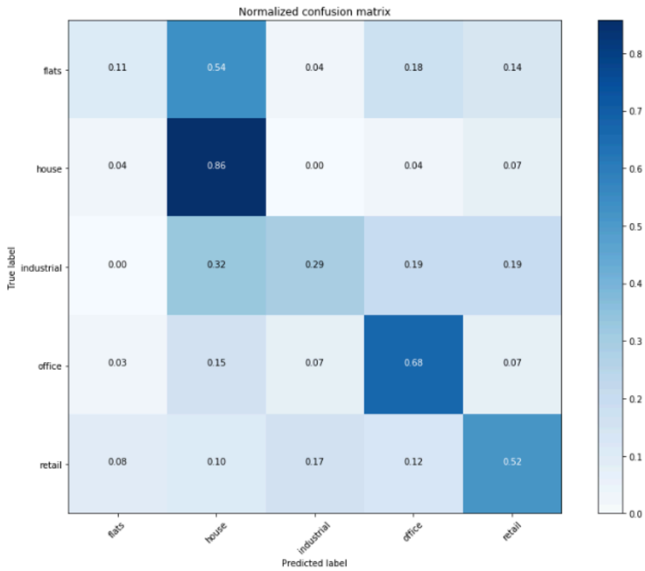


Fig 6.12 *– Normalized Confusion Matrix for Best Fit Labels demonstrating Overfitting*

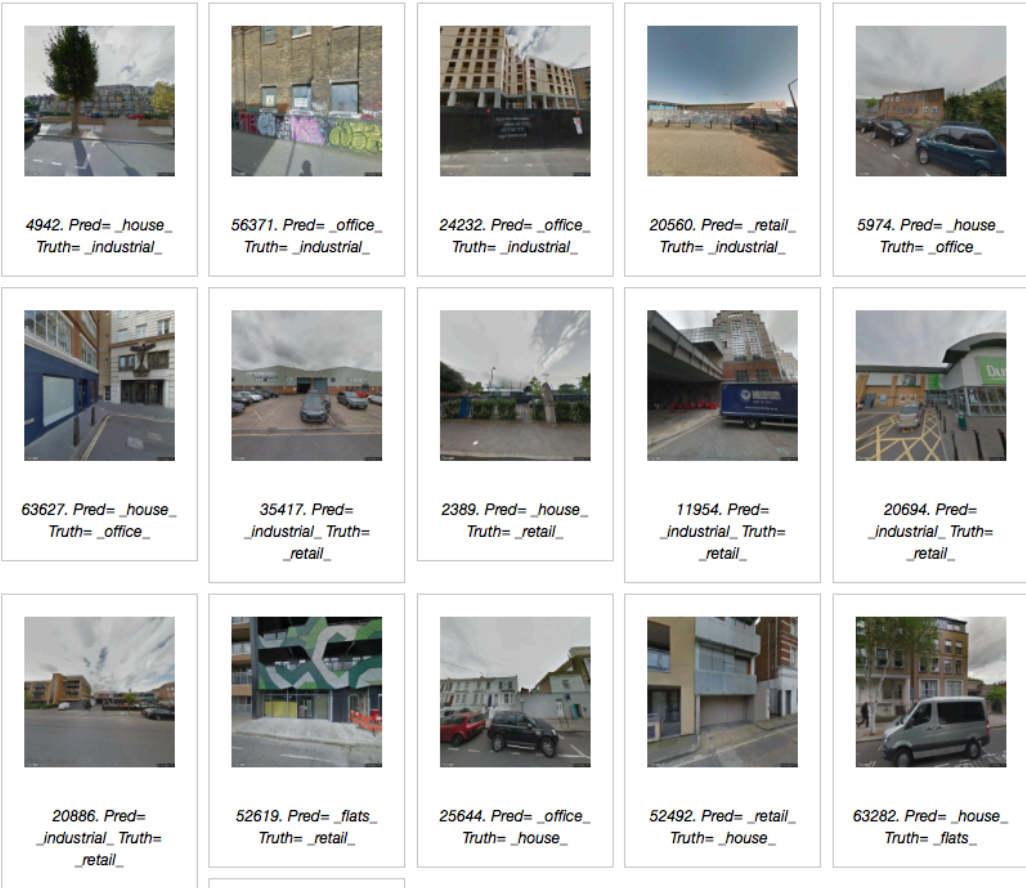


Fig 6.13 *– Thumb Gallery for Best Fit Labels Overfitting*

6.3.2 Problem Resolution Strategies

Figure 5.26 summarizes the options at our disposal for dealing with model overfit and was largely compiled after the study was completed.

The strategies and methods available fall into 3 camps, More stringent Model Record Validation and Anomaly Weeding(e.g. increasing Places 365 filter categories) , CNN Implementation Method Extension(e.g. hyper parameter tuning) and Workflow Element redesign(varying label creation controls).

Using the Tensorflow Keras api would allow us the ability to unfreeze layers, a common technique in transfer learning for gaining higher precision rates. Using Weights that had been trained on a different dataset is also another strong contender for dealing with overfitting concerns. However deploying either of these options at this point in the study was not practical due to time and ability constraints and would involve a ground up redesign of the methodology in use. We were able to deploy the former in extension work, as the Keras api was used in other areas of the study(Conv Net Visualization). Using the Places 365 PreProcessing Model weights might have been another promising potential option, though once again would involve a significant departure from the chosen methodology.

Despite this there were several other areas that warranted consideration and would nevertheless need to be discounted/eliminated as problematic factors in our particular classification task.

Figure 6.10 summarizes the range of potential factors that needed to be addressed to verify the validity of the workflow processes and elements.

The following sections describe how we varied label set creation methods, revalidated the address and model data, reexamined the pre processing workflow stage, and applied a methodical approach to considering other potential factors such as image composition and model architecture design.

We then proceed to reconsider the nature of the data itself and the tension between visual and descriptive form, building types vs. use types to identify several areas where we found the data as predisposed to being problematic for a classification exercise.

Problem	Description	Contingency/Control	Limitation	Discusssion
Development Scale	Size of Development has large impact on nature of Building Form. Leads to blurring of Building Type Set Membership. Machine Learner Image Classifiers do not like non distnct boundaries of visual type.This ambiguity in Visual Physical Urban form is accounted for by the Difference between Use Class and Building Form. For instance, Change of Use (as per The Town and Country Planning (General Permitted Development) (England) Order 2015 (as amended) (GPDO))Permission from, to take one example, Office(B1a) to Flat/Residential(C3) - for the creation of 2 seperate affordable dwelling units- above Retail Shops on Medium Scale Town Centre HIgh Street. The newly created Building Object, could be categorized, in equal measure: the Flat Use Class, Terrace Building Type, Retail Use Class, Retail Building Type, Semi Detached(if end of terrace) Building Type and Retail Building Type. From POV our Street Level Image Data	Fuzzy Logic Controls. Filter on Development Scale. Labels to focus on Building Form not Use Class.		Where do Use Class boudaries end and Building Form categories begin? For instance, an Office Blcck or a Modern Residetntial Development. If, from a Gibsonian Ecoligical Perspective, the Visual domain determines the way we think, perceieve. What is the significance of the change in the Visual Grammer - is this an oversight in the sense of Levebre's Blind Fields - Urban Phenomemen.
UnEnforced Planning Violation		N/A Data Validation Issue - If possible Remove Anomaly		Resource/Austerity. LA attitude: Laissez faire. Promote Deve(Croydon) or Preservation.
Demolition Permission		N/A Data Validation Issue - If possible Remove Anomaly		
Incomplete/UnStarted Building Permission		N/A Data Validation Issue - If possible Remove Anomaly		Land Banking.
SNN Addressing Error		N/A Data Validation Issue - If possible Remove Anomaly		
Geo Reference Issue		N/A Data Validation Issue - If possible Remove Anomaly		
Join Validation Error		See Below		
Python CSV Encoding Errror		Cross Refernce Visual Image with Address Descriptors throughout workflow		
Google Street View No Imagery		Pre Processing		
Google Street View Wrong Way	Nearest Address to Lat Lon - Assemblage			
Occluded Images		Pre Processing		
Download Error		Cross Refernce Visual Image with Address Descriptors throughout workflow		

Problem	Description	Contingency/Control	Limitation	Discusssion
Download Error		Cross Refernce Visual Image with Address Descriptors throughout workflow		
Mixed Use Policy				
Assemblage: InAccessibility				
Elevation		pov param for streetview metadata api		
Brightness		greyscale	Fairly Uniform although the 2 Batch Runs have slighlty different FOV settings which varies the amoun of sunlight in the image	
Places 365 Category Bloat		Refined and Repeated run throughs of the Pre Process Stage	Time Expensive	Places 365 Limitations
Image Subject to Environment Ratio - Feature Creep			See Section 6.3.7 - CNNs Activation found to be not be senstive to Occlusions	

Fig 6.15 *– Summary(Cont) of Potential Data Issues for our Image Classification Task higlighting the need for workflow tools that assist in troubleshooting the CNN Training and Classification stages*

6.3.3Workflow Design Mitigation - Label Creation Redesign

Our initial approach to label set creation succeeded in repesenting common varieties of machine learning label techniques but also suffered from lack of foresight and experience:

1 - Address Base Variations
Success: circa 70% Accuracy, Diverse Descriptive Detail of Building Type Cross Sections
Limitation/Oversight: Labels Not of equal Size, Tendency to Overfit

2- One vs All/ Misfit Class
Success: circa 80% Accuracy, Fast Train Run Times
Limitation: Weak Model Generalisation/Application

To correct the shortcmings of the initial workflow run throughs a second approach was taken later on, the earlier described n=200 and the n=1000 label approach. The initial runs were inadvertantly carried out using label set sizes of differing sizes, this would skew the training in favour of the more numinous set size labels. The revised approach allows for model runs that were both typological diverse but also sufficient in sample size

Training runs metrics can be found in the Table 6.12 and 6.13 below and after considering 2 further potential model overfit factors(architecture and image composition) and outlining the model validation process carried out, our final analysis of results are presented in the last section of this workflow chapter.

6.3.4 Mitigation Option Trade Offs

Figure 6.11 highlights how improvements in one problem area might cause problems for other stages of the workflow.

Workflow Problem Scenario	**Effect**	**Problem**
Improved Pre-Processing Stage	Reduction in Sample Size	Sample Size not big enough (needs to be n=1000s) and Label Diversity Reduced
Fuzzy Control – Simple Type Filter	Reduction in Sample Size (n= 100s)	Sample Size not big enough (needs to be n=1000s) and Label Diversity Reduced
Increase in Population Size	Pre-Processing Time = 1 Day on Auxiliary Workflow Laptop	Pre Processing Time
Increase in Population Size	Reduction in Label Diversity and Skewed Label Image Count	Only Flats and House Classes Combined

Fig 6.16 *– Mitigation Option Trade Offs*

Phase 2	Label Sets to deal with Poor Type Cross Sections and Over Fitting													
	Label SubSet Variation	Label Mix	Model	Platform	Location	Step Count	Number of Images	Number of Labels	Test Accuracy	Image Folder Location	Summary Folder Location	Time Taken	Comments	Notes
	Benchmark Categories:													
	Kang et al Building Instance Variations													
	Trained US OSM PROPERTY TYPES Model applied to US OSM Test Set	Apartment, Church, House, Industrial, Roof, Office, Retail, Garage	mobnet v 1.0	CPU	US Cities (Montreal, New York, Denver)	500			58.6% (N=1782)					Un-Processed
	Trained US OSM PROPERTY TYPES Model applied to US OSM Test Set	Apartment, Church, House, Industrial, Roof, Office, Retail, Garage	mobnet v 1.0	CPU	US Cities (Montreal, New York, Denver)	1000			60.7% (N=1782)					Un-Processed
	Trained US OSM PROPERTY TYPES Model applied to US OSM Test Set	Apartment, Church, House, Industrial, Roof, Office, Retail, Garage	mobnet v 1.0	CPU	US Cities (Montreal, New York, Denver)	4000			62.2% (N=1782)					Un-Processed
	Trained US OSM PROPERTY TYPES Model applied to US OSM Test Set	Apartment, Church, House, Industrial, Roof, Office, Retail, Garage	Quant Mob	CPU	US Cities (Montreal, New York, Denver)	1000			59.9% (N=1782)					
	Best Fit, Equal Size, Granular and Maximum Sample Size													
	AB as OSM CLASSES *Best FIT Label(Original Run)	Houses(Grouped), No Minor Flats, No Minor Offices, Retail, Industrial		CPU	London Test and London Train					2_3	51			
	AB as OSM CLASSES *Hybrid Types	flat_office, house, industrial		CPU										
	AB as OSM CLASSES *One vs All			CPU										
	AB as OSM CLASSES *n=250	FLAT_HOUSE_INDUSTRIAL_OFFICE		CPU		1000			60.0% (N=50)				Not Enough Training Label members & Too many Confounders	Not Fuzzy Filtered or Not Places365 Cleaned
	AB as OSM CLASSES *n=250	FLAT_HOUSE_INDUSTRIAL_RETAIL_OFFICE		CPU		1000			66.7% (N=78)					
	AB as OSM CLASSES *n=250	FLAT_HOUSE_INDUSTRIAL_RETAIL		CPU		1000			74.5% (N=51)					
	AB as OSM CLASSES *n=1000	FLAT_HOUSE_OFFICE_RETAIL		CPU										
	AB as OSM CLASSES *n=2500+ Set 1			CPU										Not Fuzzy Filtered or Not Places365 Cleaned
	AB as OSM CLASSES *n=2500+ Set 2			CPU						???				
				CPU										
	Hybrid Types (Building Rejections)													
	Equal Bin Size	house, flat, office, retail, industrial		CPU		1000			68.1%(n=47)	72	179			
	Equal Bin Size	flat_office		CPU										
	Equal Bin Size	house, flat, office, retail, industrial, TERRACES		CPU		1000			50.0% (N=78)	/Terrraces_Too	180			
	Equal Bin Size	terrace_house		CPU		1000			61.7% (N=60)	RRACES_HOUSES/	181			
		flat_house_commerrcial		CPU										
	Simple Building Types (Fuzzy Variations v 2)													
	Equal Bin Size	House, Flat, Office, Industrial, Retail		CPU										
				CPU										
	One vs Rest (1000plus)													
	Equal Bin Size	House, Flat, Office, Industrial, Retail		CPU										
				CPU										
	Misfit Classes													
				CPU										
				CPU										

Fig 6.17 *– Phase 2 – Model Development Strategies: More Training Data vs More Label Diversity*

6.3.5 Misclassified Images Workflow Tool for Model Validation

The TF Script provided a misclassified parameter option for identifying incorrect classifications. We captured these images and inspected them as part of the classification process, allowing us to verify and analyze classes of label and assess model performance. This would allow us to visualize exactly what the network was struggling with and along with the map tools verify whether the image was correctly geo-referenced and linked to the descriptive data that was been provided to the network, a process necessitated by the scale of the data model and uncertainty in the data limitations described in the methodology section.

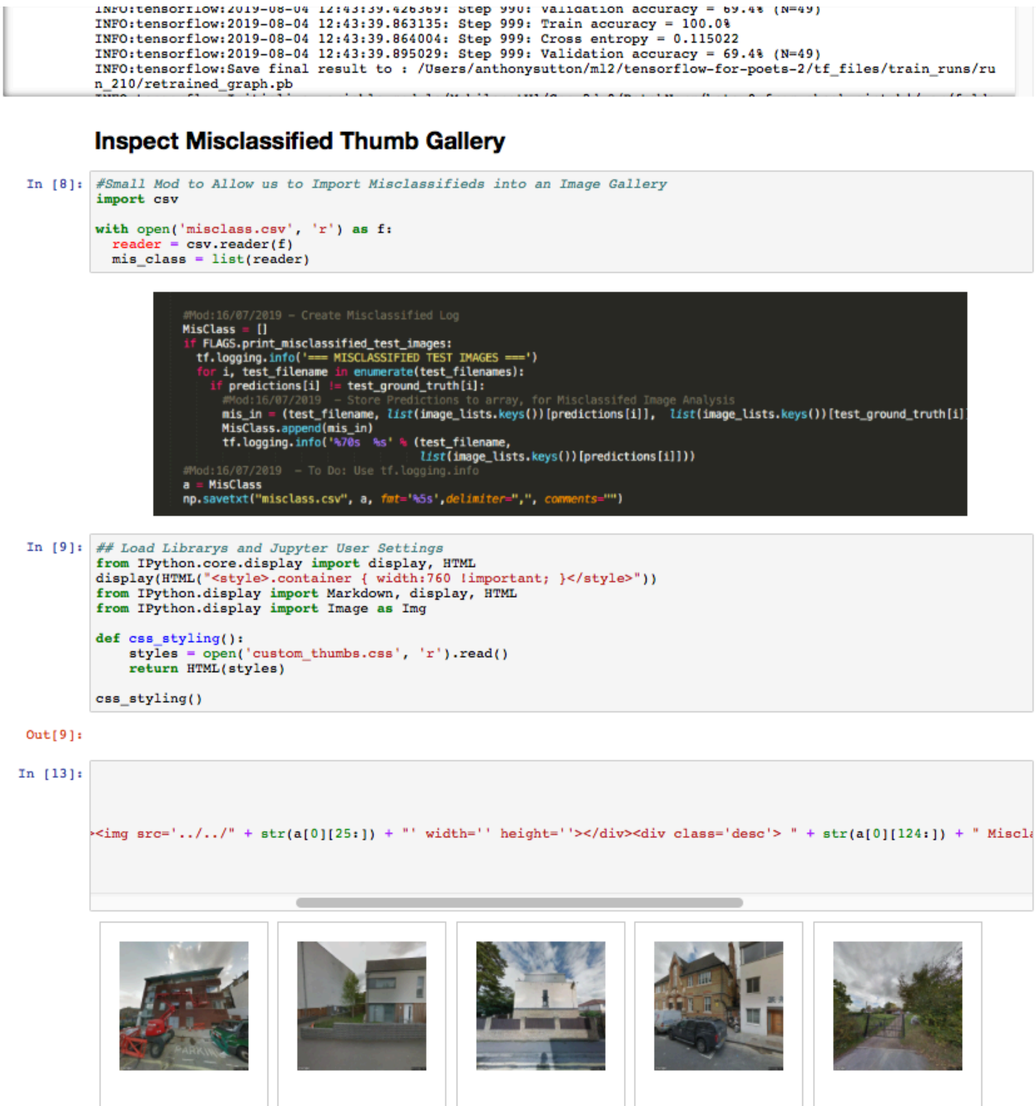


Fig 6.18 – *Script Modification for extracting Misclassified Images*

```
In [28]: #Version 3
gallery = ""

for a in mis_class:
    #print(str(a[0])[79:])
    #print(str(a[1]))
    inp3 = "<div class='gallery'><div class='zoom'><img src='../../" + str(a[0][25:]) + "' width='' height=''></d
    gallery = gallery + inp3

display(HTML(gallery))
```

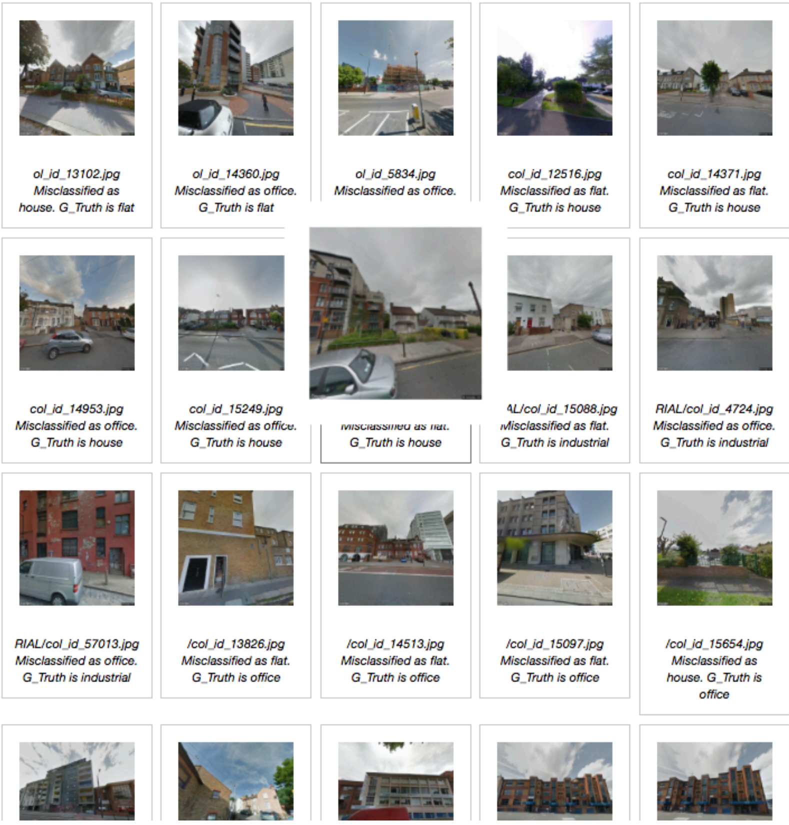
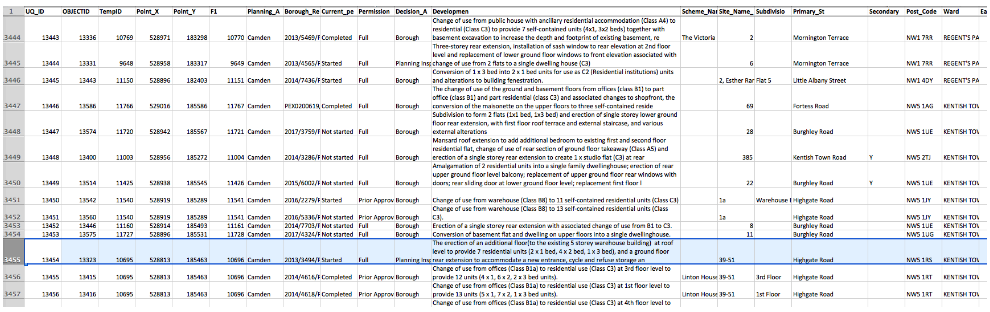


***Fig 6.19** – Thumb Gallery for MisClassified Images*



1	UQ_ID	OBJECTID	TempID	Point_X	Point_Y	F1	Planning_A	Borough_Re	Current_pe	Permission	Decision_A	Developmen	Scheme_Nar	Site_Name_	Subdivisio	Primary_St	Secondary	Post_Code	Ward	Ea
3444	13443	13336	10769	528971	183298	10770	Camden	2013/5469/F	Completed	Full	Borough	Change of use from public house with ancillary residential accommodation (Class A4) to residential (Class C3) to provide 7 self-contained units (4x1, 3x2 beds) together with basement excavation to increase the depth and footprint of existing basement, re	The Victoria	2		Mornington Terrace		NW1 7RR	REGENT'S PA	
3445	13444	13331	9648	528958	183317	9649	Camden	2013/4565/F	Started	Full	Planning Insp	Three-storey rear extension, installation of sash window to rear elevation at 2nd floor level and replacement of lower ground floor windows to front elevation associated with change of use from 2 flats to a single dwelling house (C3)		6		Mornington Terrace		NW1 7RR	REGENT'S PA	
3446	13445	13443	11150	528896	182403	11151	Camden	2014/7436/F	Started	Full	Borough	Conversion of 1 x 3 bed into 2 x 1 bed units for use as C2 (Residential institutions) units and alterations to building fenestration.		2, Esther Rar	Flat 5	Little Albany Street		NW1 4DY	REGENT'S PA	
3447	13446	13586	11766	529016	185586	11767	Camden	PEX0200619,	Completed	Full	Borough	The change of use of the ground and basement floors from offices (class B1) to part office (class B1) and part residential (class C3) and associated changes to shopfront, the conversion of the maisonette on the upper floors to three self-contained reside		69		Fortess Road		NW5 1AG	KENTISH TOV	
3448	13447	13574	11720	528942	185567	11721	Camden	2017/3759/F	Not started	Full	Borough	Subdivision to form 2 flats (1x1 bed, 1x3 bed) and erection of single storey lower ground floor rear extension, with first floor roof terrace and external staircase, and various external alterations		28		Burghley Road		NW5 1UE	KENTISH TOV	
3449	13448	13400	11003	528956	185272	11004	Camden	2014/3286/F	Not started	Full	Borough	Mansard roof extension to add additional bedroom to existing first and second floor residential flat, change of use of rear section of ground floor takeaway (Class A5) and erection of a single storey rear extension to create 1 x studio flat (C3) at rear		385		Kentish Town Road	Y	NW5 2TJ	KENTISH TOV	
3450	13449	13514	11425	528938	185545	11426	Camden	2015/6002/F	Not started	Full	Borough	Amalgamation of 2 residential units into a single family dwellinghouse; erection of rear upper ground floor level balcony; replacement of upper ground floor rear windows with doors; rear sliding door at lower ground floor level; replacement first floor l		22		Burghley Road	Y	NW5 1UE	KENTISH TOV	
3451	13450	13542	11540	528919	185289	11541	Camden	2016/2279/F	Started	Prior Approv	Borough	Change of use from warehouse (Class B8) to 11 self-contained residential units (Class C3)		1a	Warehouse E	Highgate Road		NW5 1JY	KENTISH TOV	
3452	13451	13560	11540	528919	185289	11541	Camden	2016/5336/F	Not started	Prior Approv	Borough	Change of use from warehouse (Class B8) to 13 self-contained residential units (Class C3).		1a		Highgate Road		NW5 1JY	KENTISH TOV	
3453	13452	13446	11160	528914	185493	11161	Camden	2014/7703/F	Not started	Full	Borough	Erection of a single storey rear extension with associated change of use from B1 to C3.		8		Burghley Road		NW5 1UE	KENTISH TOV	
3454	13453	13575	11727	528896	185531	11728	Camden	2017/4324/F	Not started	Full	Borough	Conversion of basement flat and dwelling on upper floors into a single dwellinghouse.		11		Burghley Road		NW5 1UG	KENTISH TOV	
3455	13454	13323	10695	528813	185463	10696	Camden	2013/3494/F	Started	Full	Planning Insp	The erection of an additional floor(to the existing 5 storey warehouse building) at roof level to provide 7 residential units (2 x 1 bed, 4 x 2 bed, 1 x 3 bed), and a ground floor rear extension to accommodate a new entrance, cycle and refuse storage an		39-51		Highgate Road		NW5 1RS	KENTISH TOV	
3456	13455	13415	10695	528813	185463	10696	Camden	2014/4616/F	Completed	Prior Approv	Borough	Change of use from offices (Class B1a) to residential use (Class C3) at 3rd floor level to provide 12 units (4 x 1, 6 x 2, 2 x 3 bed units).	Linton House	39-51	3rd Floor	Highgate Road		NW5 1RT	KENTISH TOV	
3457	13456	13416	10695	528813	185463	10696	Camden	2014/4618/F	Completed	Prior Approv	Borough	Change of use from offices (Class B1a) to residential use (Class C3) at 1st floor level to provide 13 units (5 x 1, 7 x 2, 1 x 3 bed units).	Linton House	39-51	1st Floor	Highgate Road		NW5 1RT	KENTISH TOV	
												Change of use from offices (Class B1a) to residential use (Class C3) at 4th floor level to								

***Fig 6.20** – Matching a Visual inspection of the Image to its Labels via the Data Model.*

Outline of Labe-Image-Building Record Validation Process:

1- Run Train Model
2- Inspect Accuracy/Loss Curves
3- Inspect Inline or Classify Script Confusion Tables
4- Load Misclassified File into Thumb Gallery Inspector
5- View Prediction and Ground Truth Labels.
6- Check Record correctly matched in Master Data Model

6.3.6 What the CNN “Sees”

Understanding how a Convolutional Net reaches its trained classification decision can give us an insight into whether there is a potential issue with a given model architecture and/or the class and sample of images from our data model.

The two-dimensional filters learned by the model can be inspected and visualized to discover the types of features that the model will detect, and the activation maps output by convolutional layers can be inspected to understand exactly what features were detected for a given input image.

We look at two popular methods: raw visualization of the convolutional layers and class activation heat maps(CaM). We use the VGGModel to highlight general points about CNN activations and proceed to examine visualize the activations in use by the mobilenet architecture which based on Inception that we predominately use in the workflow.

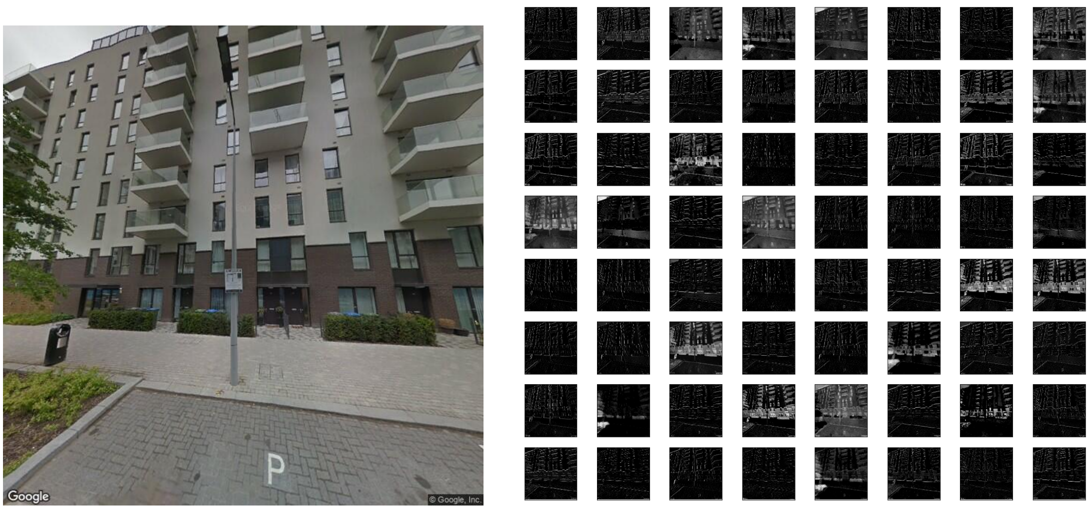


Fig 6.21 *– On the left is our Target Image(Flat Building Type) and to the right as seen through a VGG16 Model convolutional layer filter*

Layer Visualization

The activation maps, called feature maps, capture the result of applying the filters to input, such as the input image or another feature map.

The idea of visualizing a feature map for a specific input image would be to understand what features of the input are detected or preserved in the feature maps. The expectation would be that the feature maps close to the input detect small or fine-grained detail, whereas feature maps close to the output of the model capture more general features.

We can use this information and design a new model that is a subset of the layers in the full VGG16 model. The model would have the same input layer as the original model, but the output would be the output of a given convolutional layer, which we know would be the activation of the layer or the feature map.

There are five main blocks in the image (e.g. block1, block2, etc.) that end in a pooling layer. we can now create five separate plots for each of the five blocks in the VGG16 model

We can define a new model that has multiple outputs, one feature map output for each of the last convolutional layer in each block. we can cap the number of feature maps visualized at 64 for consistency.

```
1 block1_conv1 (?, 224, 224, 64)
2 block1_conv2 (?, 224, 224, 64)
4 block2_conv1 (?, 112, 112, 128)
5 block2_conv2 (?, 112, 112, 128)
7 block3_conv1 (?, 56, 56, 256)
8 block3_conv2 (?, 56, 56, 256)
9 block3_conv3 (?, 56, 56, 256)
11 block4_conv1 (?, 28, 28, 512)
12 block4_conv2 (?, 28, 28, 512)
13 block4_conv3 (?, 28, 28, 512)
15 block5_conv1 (?, 14, 14, 512)
16 block5_conv2 (?, 14, 14, 512)
17 block5_conv3 (?, 14, 14, 512)
```

```
# visualize feature maps output from each block in the vgg model
from tensorflow.keras.applications.vgg16 import VGG16
from tensorflow.keras.applications.vgg16 import preprocess_input
from tensorflow.keras.preprocessing.image import load_img
from tensorflow.keras.preprocessing.image import img_to_array
from tensorflow.keras.models import Model
from matplotlib import pyplot
from numpy import expand_dims
# load the model
model = VGG16()
# redefine model to output right after the first hidden layer
ixs = [2, 5, 9, 13, 17]
outputs = [model.layers[i].output for i in ixs]
model = Model(inputs=model.inputs, outputs=outputs)
```

Fig 6.22 *– The layer indexes of the last convolutional layer in each block are [2, 5, 9, 13, 17].*

***Fig 6.23** – Target Image for the Conv Net Layer Filter and CaM Visualisation Inspection*

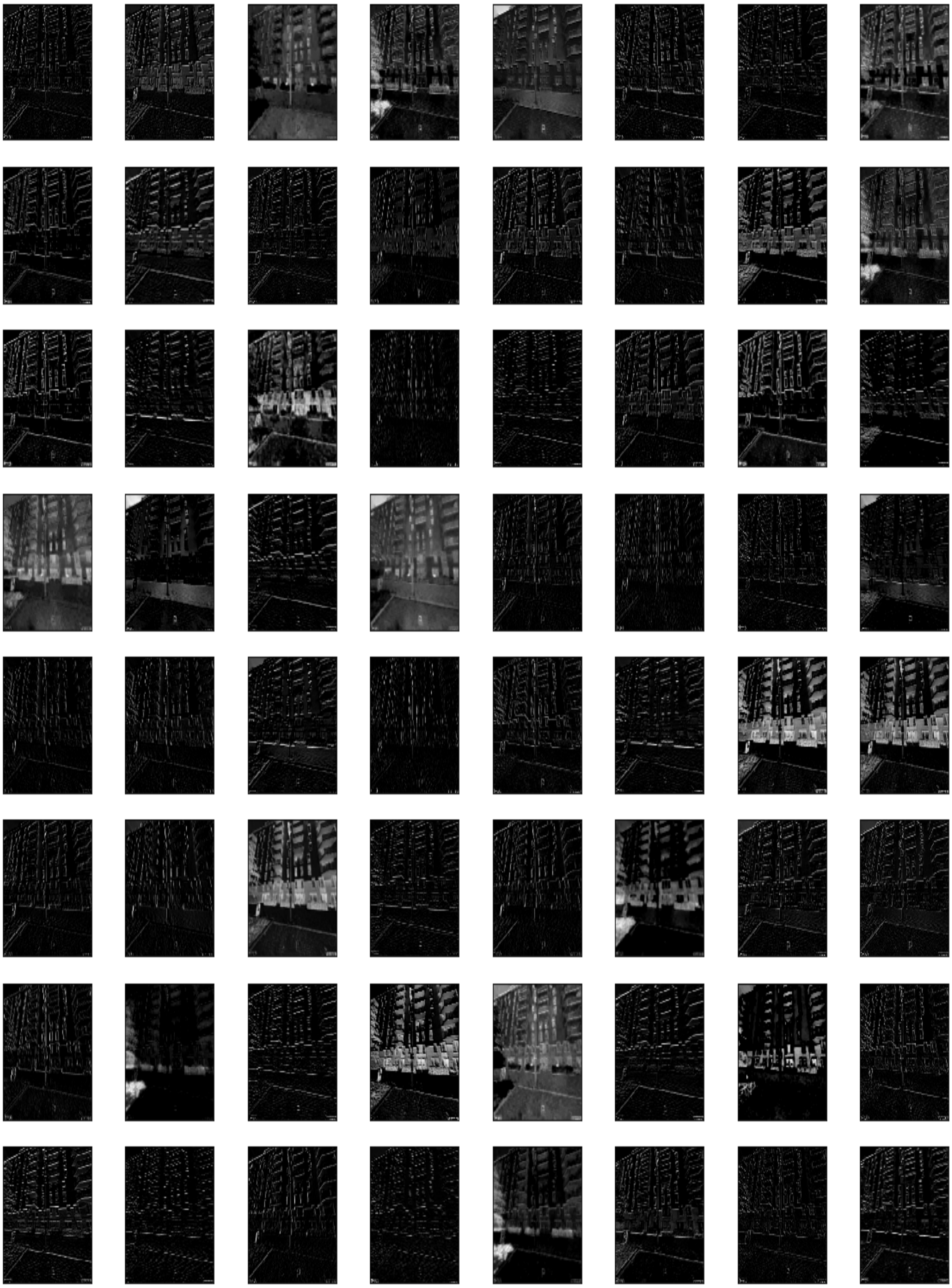


Fig 6.24 *– Feature Maps of Layer Index 2 of the VGG Model*

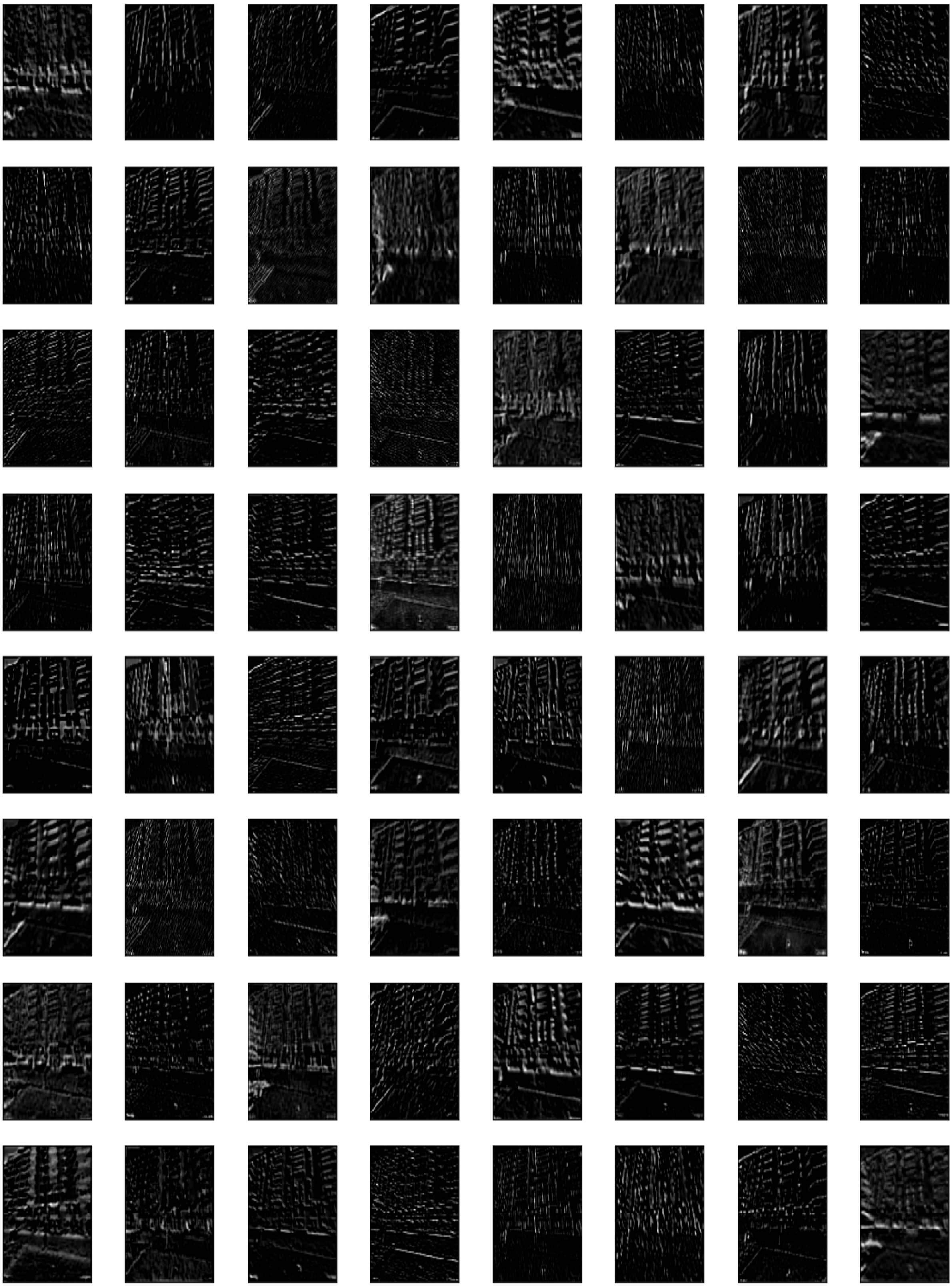


Fig 6.25 *– Feature Maps of Layer Index 5 of the VGG Model*

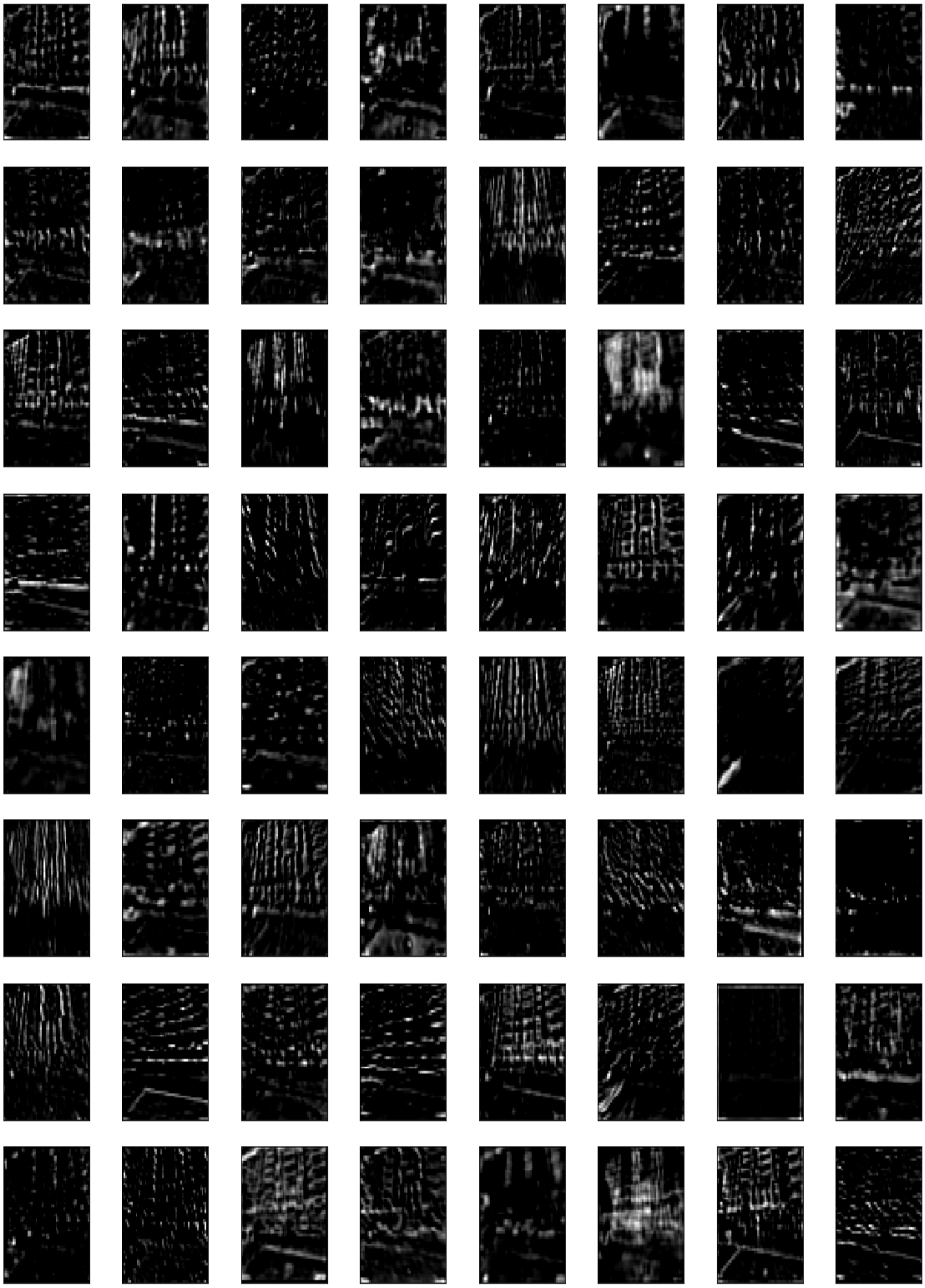


Fig 6.26 *– Feature Maps of Layer Index 9 of the VGG Model*

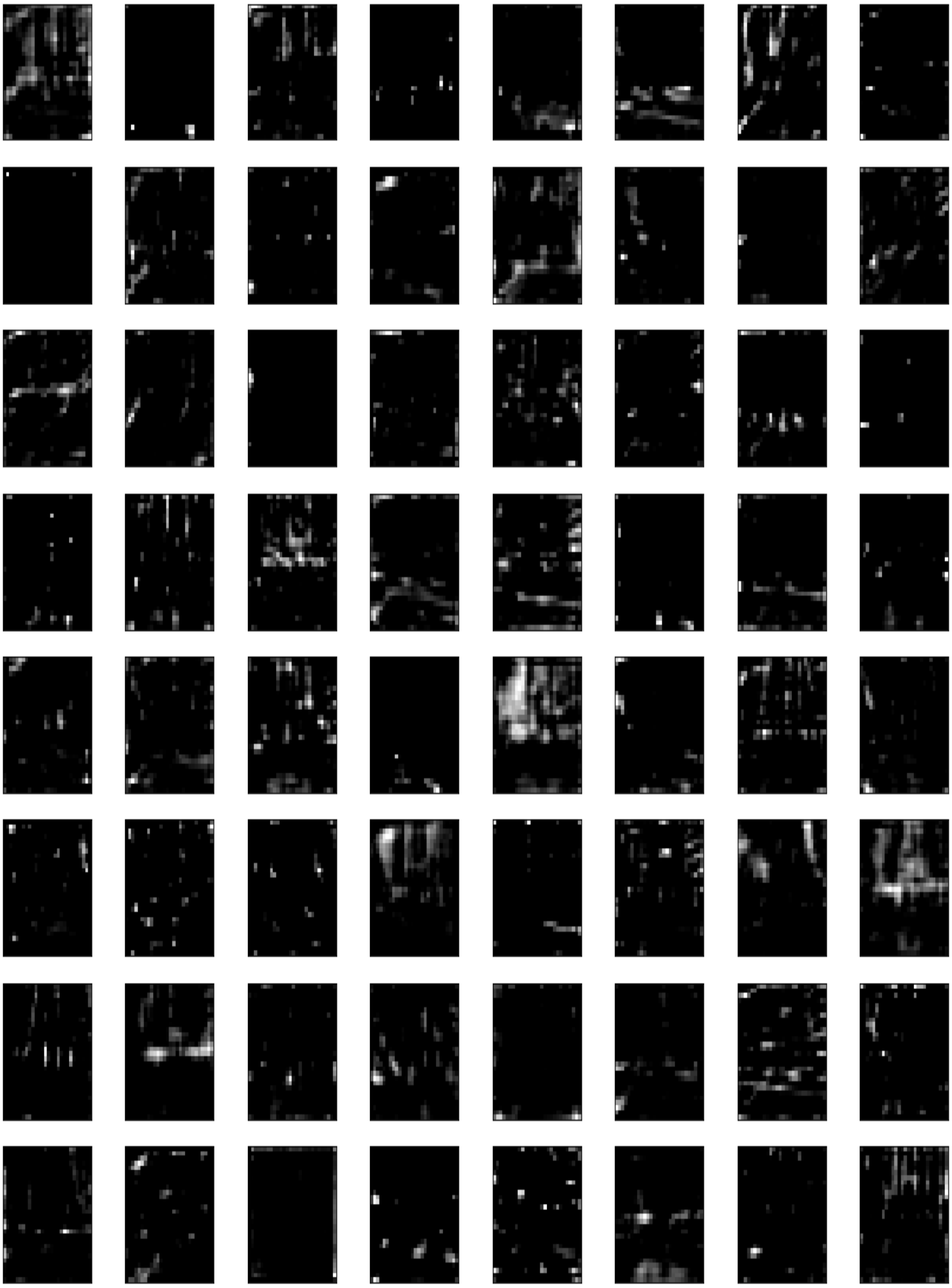


Fig 6.27 *– Feature Maps of Layer Index 13 of the VGG Model*

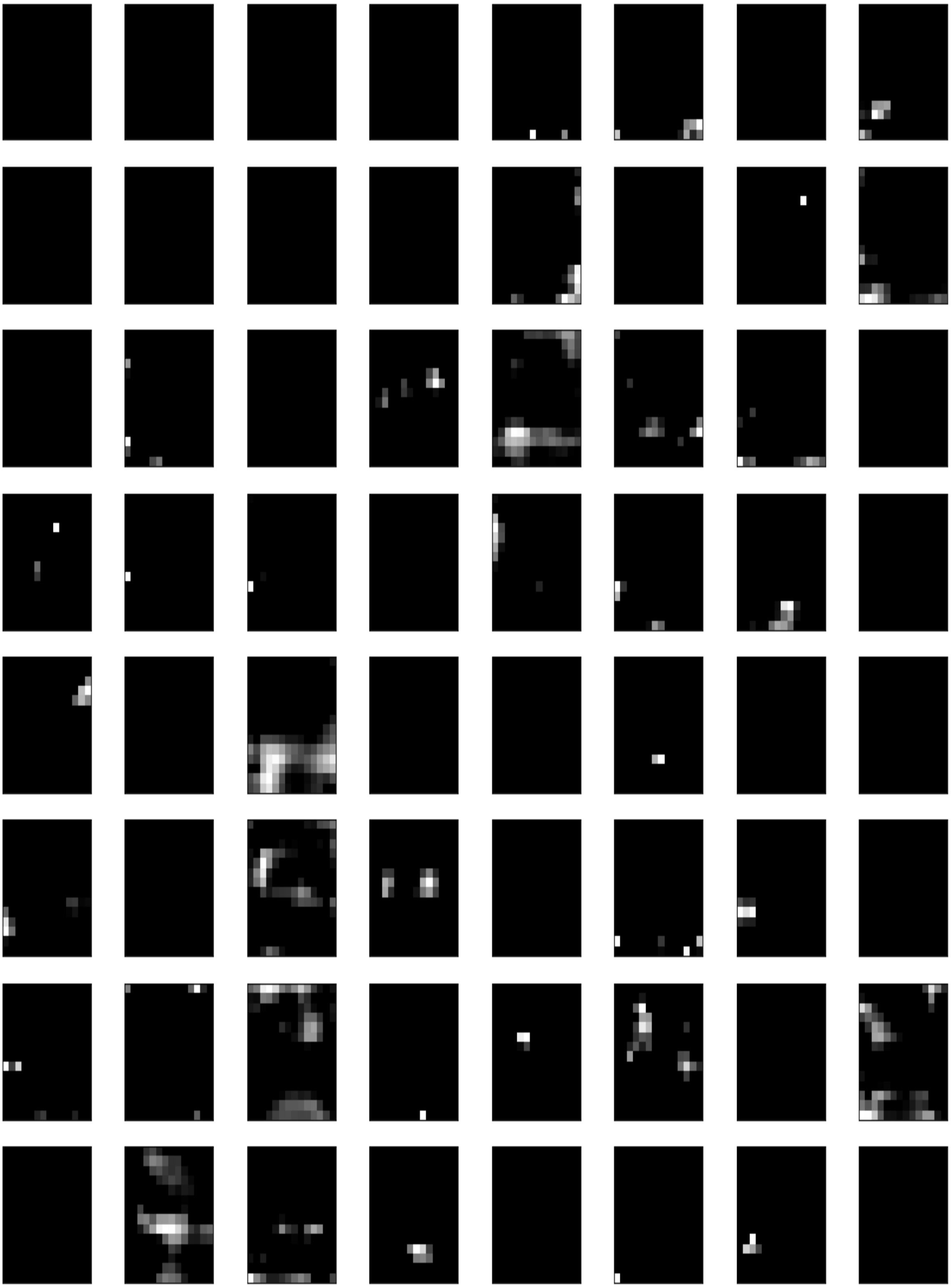


Fig 6.28 *– Feature Maps of Layer Index 17 of the VGG Model*

We can see that the feature maps closer to the input of the model capture a lot of fine detail in the image and that as we progress deeper into the model, the feature maps show less and less detail. Layers that are deeper in the network visualize more training data specific features, while the earlier layers tend to visualize general patterns like edges, texture, background etc.

This pattern is to be expected, it becomes very clear what different layers are actually trying to learn from the image data provided to them. The patterns found in filters in starting layers seem to be very basic, composed of lines and other basic shapes, which tell us that the earlier layers learn about basic features in images like edges, colors, etc. But as you move deeper into the network, the patterns get more complex, suggesting that the deeper layers are actually learning about much more abstract information, which helps these layers to generalize about the classes and not the specific image. And this is why, we saw a few empty filters activations in deeper layers in the previous section, because that particular filter was not activated for that image, in other words, the image does not have the information that the filter was interested in. The reason for this is not the incorrectness of the network, but rather the larger fraction of image that is occupied by other objects or activation features.

Class Activation Map(CAM) and Grad Cam Visualizations

Fig 6.29 *– Target Images for the CAM Viz Inspection. Moving Clockwise: True Office, True Flat, Office Predicted as House and Office Predicted as Flat*

CAM

Class activation maps are a technique for identifying discriminative image regions used by a given CNN to identify a specific class in the image. A class activation map (CAM) lets us see which regions in the image were relevant to this class. Using the gradients of the target concept (Selvaru et al) flowing into the final convolutional layer we are able to produce a coarse localization map highlighting the important regions in the image for the CNN model prediction under observation.

Grad CAM

Instead of using gradients with respect to output (see saliency), grad-CAM uses penultimate (pre Dense layer) Conv layer output. The idea is to use the nearest Conv layer to utilize spatial information that gets completely lost in Dense layers. Grad- CAM is applicable to a wide variety of CNN model-families.

Within the context of image classification models, grad cam visualizations (a) lend insights into their failure modes (showing that seemingly unreasonable predictions have reasonable explanations), (b) are robust to adversarial images, (c) outperform previous methods on weakly-supervised localization, (d) are more faithful to the underlying model and (e) help achieve generalization by identifying dataset bias(Selvaru et al).

Example 1 - True flat

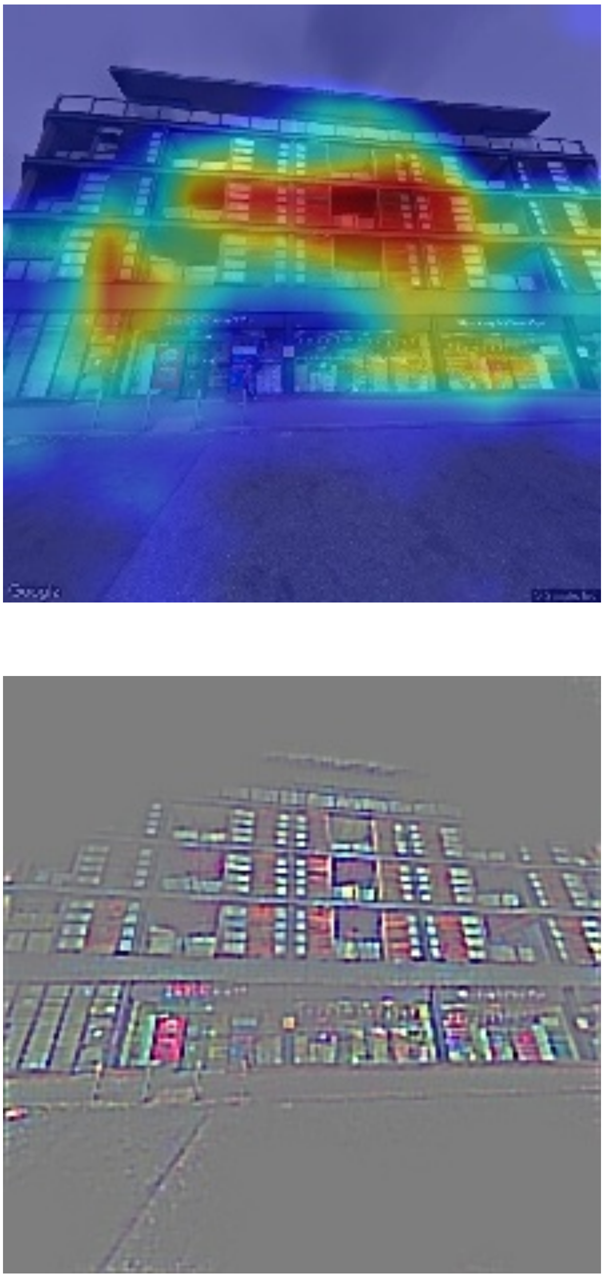
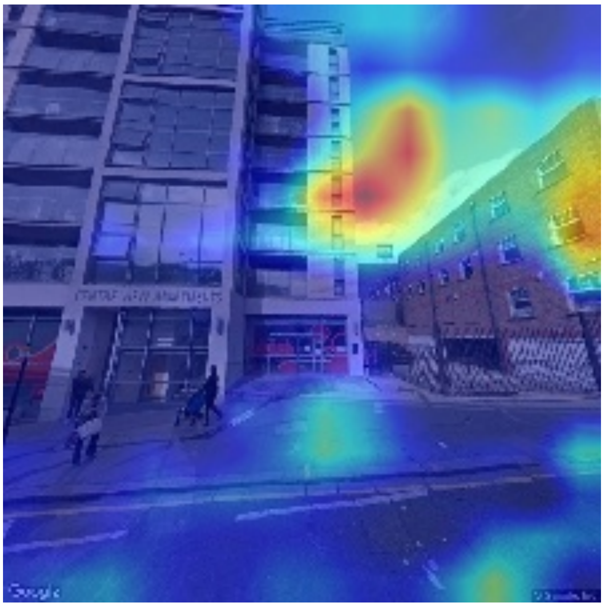


Fig 6.30 *– VGG16 network in keras.applications heat map(top) produced with from the guided grad cam activation map(bottom). In our Clasification task, this was correctly predicted as Flat however notice the mixed retail usage present on the ground floor.*

Example 2 - True office



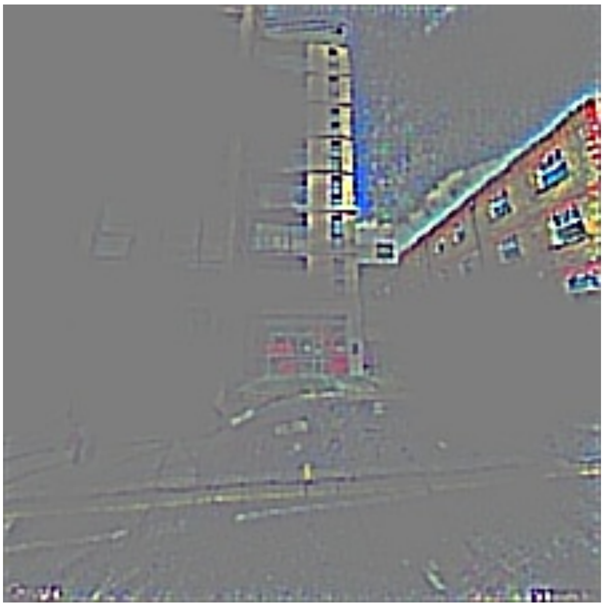


Fig 6.31 *– VGG16 network in keras.applications heat map (top) produced with from the guided grad cam activation map(bottom). In our Classification task, this was correctly predicted as Office. Notice the stark panelling and presence of additonal buildings activation ares, suggesting the location of a business zone might be a factor in this prediction.*

Example 3 - Office predicted as flat

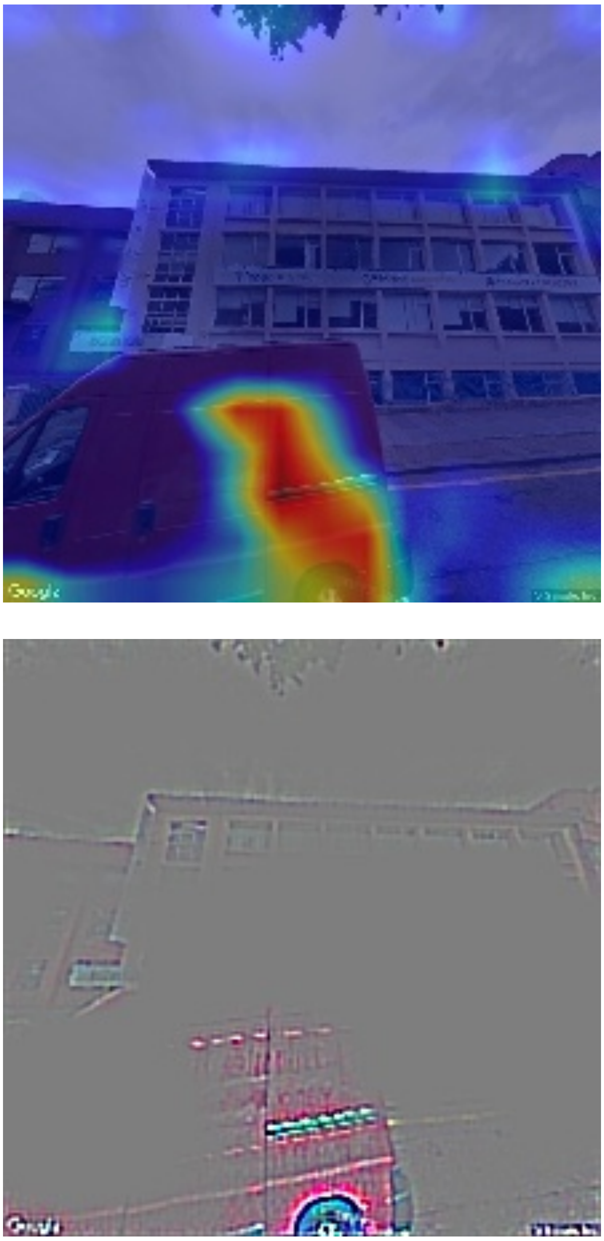


Fig 6.32 *– VGG16 network in keras.applications heat map (top) produced with from the guided grad cam activation map(bottom). In our Clasification task, this was incorrectly predicted as Flat. It was an office. The passing van in the foreground suggests this occluding factor is overwhelming themodel from making a correct predcition. Ideally this image would have been picked up in the pre processing stage.*

Example 4 - Office predicted as house

Fig 6.33 *– VGG16 network in keras.applications heat map(top) produced with from the guided grad cam activation map(bottom). In our Classification task, this was incorrectly predicted as house(via the terrace buidling type. This highlights the models problem in dealing with commonly encountered mixed high street terrace building types.*

Mobilenet – and ELI5, Python ML Debugging Visualization Library

ELI5 is a Python package which helps to debug machine learning classifiers and explain their predictions. It allows us to readily deploy Grad Cam visualizations for the MobileNet model architecture, and which along with ImageNet(on which MobileNet is based) featured predominantly in the studies workflow.

Fig 6.34 *– MobileNet produced* ELI5 *heat map. In our Clasification task, this was incorrectly predicted as house. Notice the neutral frontage and high street terrace activation errors. The eaves in particular are strongly suggestive of a house building type.*

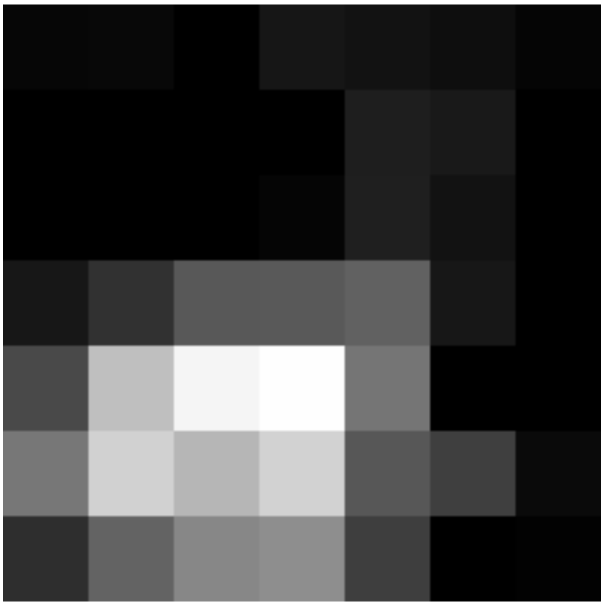


Fig 6.35 *– MobileNet produced* ELI5 *heat map and pillow image plot(top). In our Clasification task, this was incorrectly predicted as flat. These examples suggest the VGG Model activation visualisations also apply to the mobilenet CNN prediction criteria.*

```
In [30]: from PIL import Image
         from IPython.display import display
         import numpy as np

         # you may want to keep logging enabled when doing your own work
         import logging
         import tensorflow as tf
         #tf.get_logger().setLevel(logging.ERROR)
         import warnings
         warnings.simplefilter("ignore")
         import keras
         from keras.applications import mobilenet_v2

         import eli5
```

And load our image classifier (a light-weight model from `keras.applications`).

```
In [89]: model = mobilenet_v2.MobileNetV2(include_top=True, weights='imagenet', classes=1000)
```

```
for l in ['block_2_expand', 'block_9_expand', 'Conv_1']:
    print(l)
    display(eli5.show_prediction(model, doc, layer=l)) # we pass the layer as an argument
```

block_2_expand

***Fig 6.36** – ELI5 Code for Explaining Keras image classfiier predictions for MobileNet Model architecture mobilenetv2_1.00_224*

Summary of Conv-Net Visualization Result Summary

Whilst we have only scratched the surface of our visual validation of layer activation features involved and far from being conclusive, the above activations seem to suggest that nuance in colour and cosmetic surface detail that is accompanied with no major change in building form outline(as commonly found in retail signage distinctions in high street and suburban terrace building types or variation in frontal building apparel that only mildly differentiates between office and flat building types in many new build development examples) may be creating adversarial image training data examples for our CNN model.

6.3.7 Street Level Image Composition

The first asset of a high-quality dataset is an expansive coverage of the categorical space to be learned. If there are inconsistencies in the framing and field of view of our image set, then this could be an additional factor in combating model overfitting.

`fov` (*default is* `90`) determines the horizontal field of view of the image. The field of view is expressed in degrees, with a maximum allowed value of `120`. When dealing with a fixed-size viewport, as with a Street View image of a set size, field of view in essence represents zoom, with smaller numbers indicating a higher level of zoom

Occlusion Sensitivity

With image classification approaches, a natural question is if the model is truly identifying the location of the object in the image, or just using the surrounding context.

Zeiler and Fergus in their paper, 'Visualizing and understanding Convolutional Networks' demonstrated through a series of occlusion experiments on a model trained on the ImageNet dataset, that the model, while trained for classification, is highly sensitive to local structure in the image and is not just using broad scene context. Their examples clearly show the model is localizing the objects within the scene, as the probability of the correct class drops signifi- cantly when the object is occluded.

Mitigation Technique and Outcome Summary Table

Workflow Issue	Mitigation Option
Occlusions	Pre Processing
Assemblage -Private Interface	Fuzzy Control, Pre Processing
Mixed Use	Fuzzy Control, Pre Processing
Model OverFit	See Below

Fig 6.37 *– Mitigation Technique and Outcome Summary Table*

	Model OverFit Mitigation Option	Outcome	Notes
Workflow Design	Better Pre Processing	No Sig Improvement.	Kang et Al Image Datset would also include tree Oclusions
	Dynamic Field of View Setting Image Download	Not Attempted	e.g select ground floor only field of view for retail property type
	Remove Misclassified and Run Train Session Again	No Sig Improvement.	Data Quantity Reduced
ML Related	Unfreeze and Update Pre Trained NN Weights	No Sig Improvement.	
	Add weight regularization.	Not Attempted	Suitable for Trained from Scratch Model
	Add dropout	Not Attempted	Suitable for Trained from Scratch Model
	Get more training data.	Not Attempted	
	Semantic Segmentation		describes the process of associating each pixel of an image with a class label
	Use Another Base Dataset	Not Attempted	Places 365 Scene Recognition Data, Mappillary Image Segmenetation Proprietary DataSet?

Fig 6.38 *– Mitigation Technique and Outcome Summary Table*

6.4.4 Fine Tuning with Keras

In order to explore methods for model overfit mitigation, we used the Keras platform implementation of the MobileNet CNN model trained on our best fit label set selections and on the benchmark performance datasets(tf flowers and the Kang et al building instance data). tf.keras is TensorFlow's implementation of the Keras API specification. This is a high-level API to build and train models that makes it easier to use TensorFlow-specific functionality, such as eager execution, tf.datapipelines, and Estimators.

6.4.4.1 - Benchmark(TF Flowers)

```
[105] history = model.fit(train_dataset,
                          epochs=10, steps_per_epoch=115,
                          validation_data=val_dataset)

Train for 115 steps, validate for 700 steps
Epoch 1/10
115/115 [==============================] - 1620s 14s/step - loss: 0.7819 - accuracy: 0.7082 - val_loss: 0.8644 - val_
Epoch 2/10
115/115 [==============================] - 1306s 11s/step - loss: 0.4386 - accuracy: 0.8418 - val_loss: 0.7215 - val_
Epoch 3/10
115/115 [==============================] - 1289s 11s/step - loss: 0.3585 - accuracy: 0.8783 - val_loss: 0.6968 - val_
Epoch 4/10
115/115 [==============================] - 1295s 11s/step - loss: 0.3265 - accuracy: 0.8826 - val_loss: 0.6460 - val_
Epoch 5/10
115/115 [==============================] - 1312s 11s/step - loss: 0.2970 - accuracy: 0.8989 - val_loss: 0.6854 - val_
Epoch 6/10
115/115 [==============================] - 1316s 11s/step - loss: 0.2697 - accuracy: 0.9101 - val_loss: 0.5459 - val_
Epoch 7/10
115/115 [==============================] - 1305s 11s/step - loss: 0.2703 - accuracy: 0.9030 - val_loss: 0.6891 - val_
Epoch 8/10
115/115 [==============================] - 1304s 11s/step - loss: 0.2455 - accuracy: 0.9139 - val_loss: 0.7313 - val_
Epoch 9/10
115/115 [==============================] - 1315s 11s/step - loss: 0.2320 - accuracy: 0.9228 - val_loss: 0.7251 - val_
Epoch 10/10
115/115 [==============================] - 1312s 11s/step - loss: 0.2161 - accuracy: 0.9280 - val_loss: 0.6869 - val_
```

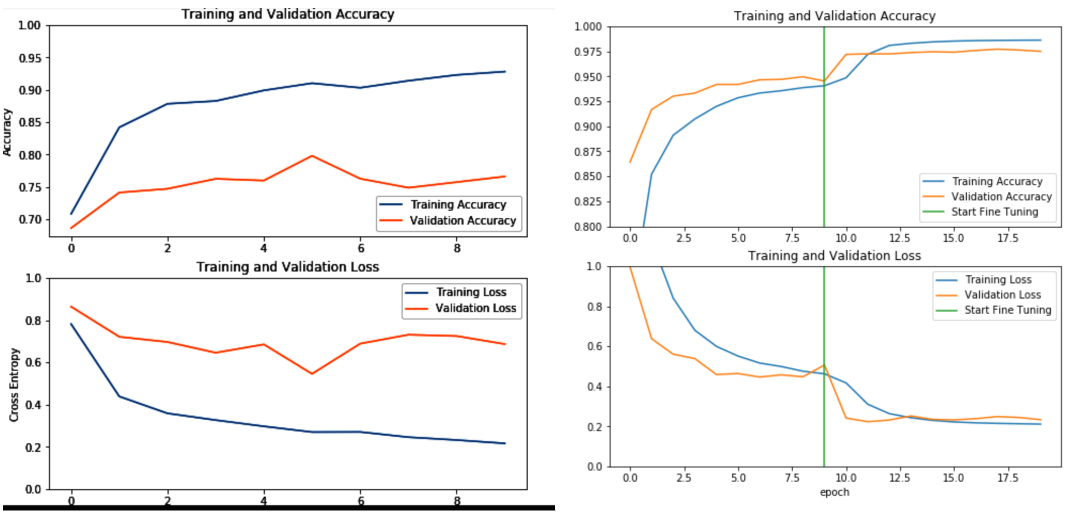


Fig 6.39 *– Keras Implementation Training and Validation*

6.4.4.2 - Best Fit Label Run 2.3 (Flat, House, Industrial, Office, Retail)

2.1 Multi Class - Training Run

```
In [4]: label_names = sorted(item.name for item in data_root.glob('*/') if item.is_dir())
        label_names

Out[4]: ['FLAT', 'HOUSE', 'RETAIL']
```

```
model.compile(optimizer=tf.keras.optimizers.RMSprop(lr=0.0001),
              loss='categorical_crossentropy',
```

```
Epoch 9/10
23/23 [==============================] - 127s 6s/step - loss: 0.7837 - acc: 0.6727 - val_loss: 1.0716 - val_acc: 0.60
62
Epoch 10/10
23/23 [==============================] - 124s 5s/step - loss: 0.7449 - acc: 0.7074 - val_loss: 1.0433 - val_acc: 0.61
87
```

Fig 6.40 *– Keras Implementation Model Compilation*

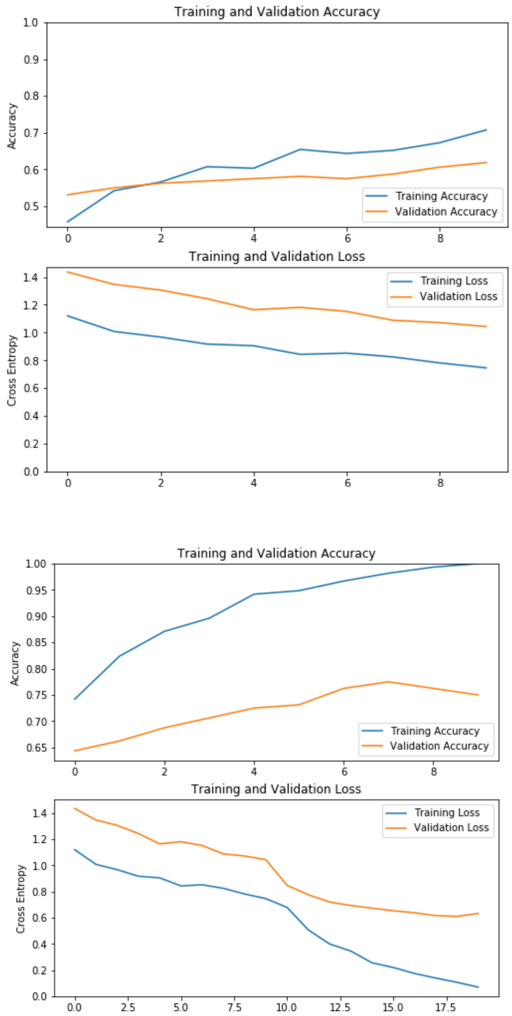


Fig 6.41 *– Keras Implementation Training and Validation Curves*

2.2 Binary Class Run Training Run

```
In [6]: label_names = sorted(item.name for item in data_root.glob('*/') if item.is_dir())
        label_names

Out[6]: ['FLAT', 'HOUSE']
```

```
model.compile(optimizer=tf.keras.optimizers.RMSprop(lr=0.0001),
              loss='binary_crossentropy',
```

Fig 6.42 *– Keras Implementation Model Compilation*

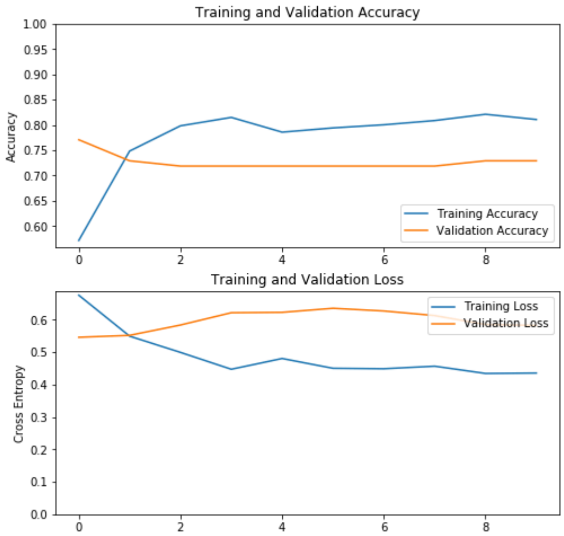
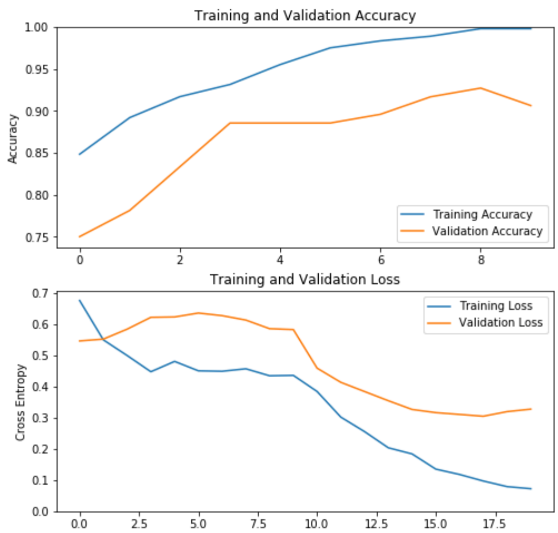


Fig 6.43 *– Keras Implementation Training and Validation Curves*



6.4.1 Analysis of Results

Kang et Al Study Comparison

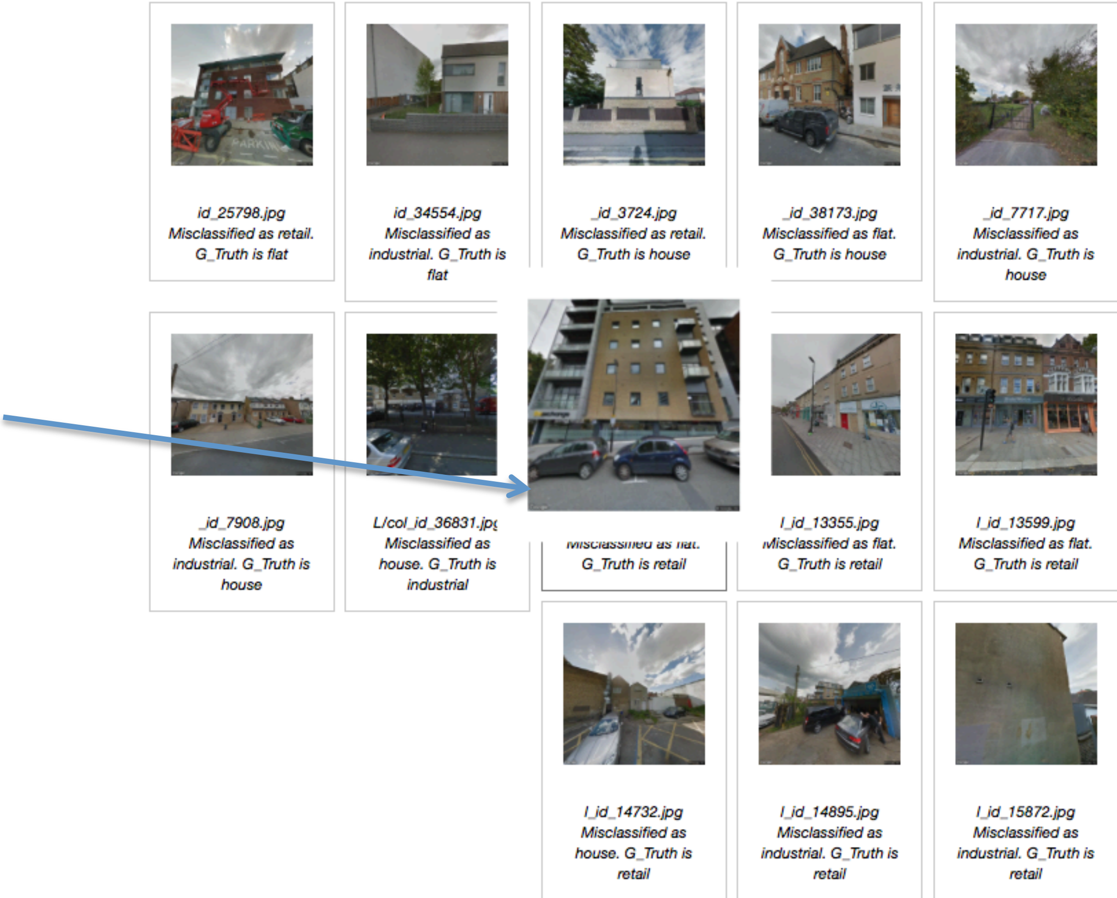


Fig 6.44 *– Misclassified Image Gallery Excerpt for Model Run on the Kang et Al US Building Instances DataSet*

In the highlighted image example we see a common confounded classification in our workflow. The predicted type is Office however the Ground Truth is found to be Retail. A typical London Mixed Use New Development, with Shops Underneath serving the new local inhabitants dwelling above, and the surrounding consumer catchment area. In the fore font traffic occludes the Supermarket Signage and Bright Colors, however the Street View photograph features the full height of the building nevertheless.

Recreating the Kang et al research on their research data which was released to the public. We see a similar pattern or misclassification. Below Office and

Apartment categories are confounded by a Modern angular building that is cropped into a close up shot of a busy downtown urban scene.

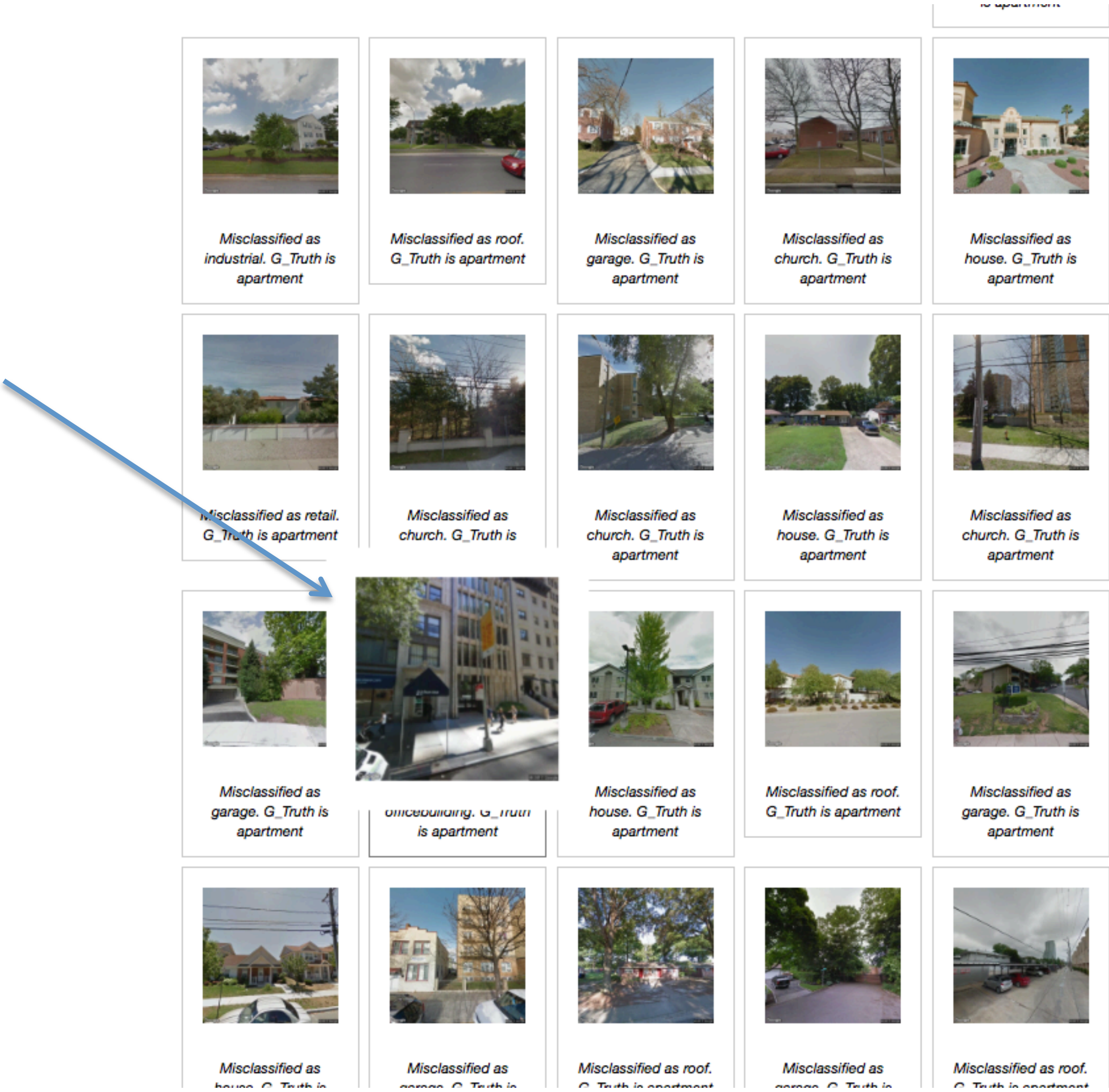


***Fig 6.45** – Misclassified Image Gallery Exceprt or Model Run on the Kang et Al US Building Instances DataSet*

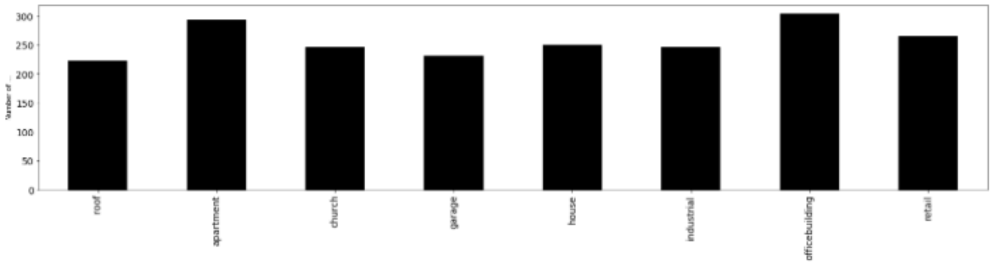
If we inspect the accuracy and loss curve for this training run, we can see no obvious sign of over fitting. The Validation run continues to improve on both directions thoughout all steps:

Fig 6.46 *– Accuracy and loss curve for Model Run on the Kang et Al US Building Instances DataSet*

This Label Set comprises of 8 label features consisting of n=25k images per label. In both critical areas(label diversity and sample count) the difference between the LDD-Google StreetView dataset is significant. This suggests the biggest weakness in our data model and the two areas to focus on to overcome model overfitting are **granular diversity** and **sample size**.

Ground Truth Property Type Distribution

In [94]:
```
#Check Distribution of Property Types FOR GROUND TRUTH
plotdat(df_ldd_label_join,'C_Truth')
```



CNN PREDICTION Property Type Distribution

In [95]:
```
#Check Distribution of Property Types FOR CLASSFIER
plotdat(df_ldd_label_join,'Predicted_Label')
```

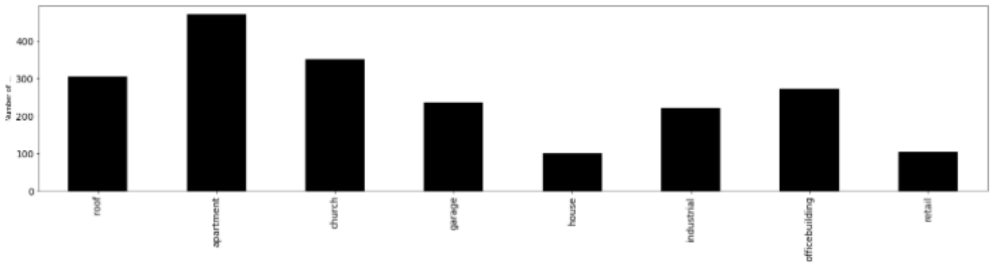
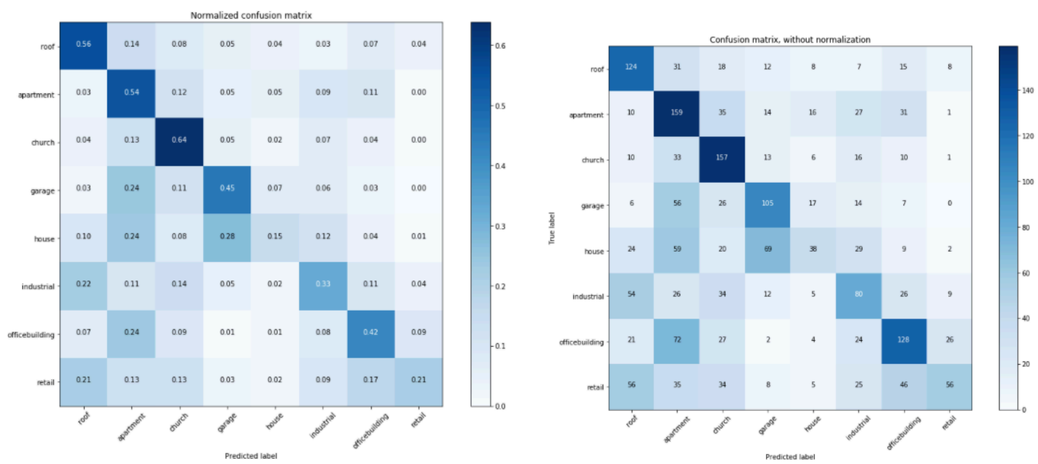


__Fig 6.47__ – Ground Truth vs CNN Prediction Scores for Model Run on the Kang et Al US Building Instances DataSet

__Fig 6.48__ – Confusion Matrix for Model Run on the Kang et Al US Building Instances DataSet



The low normalized percentage recall rate is due to this run having not been through the pre processing stage

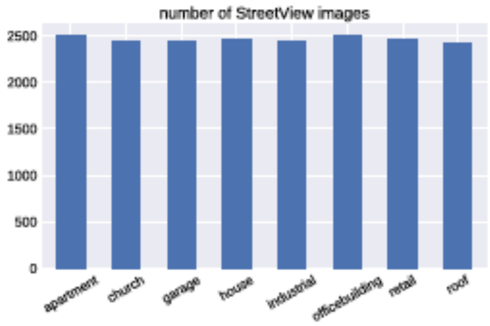
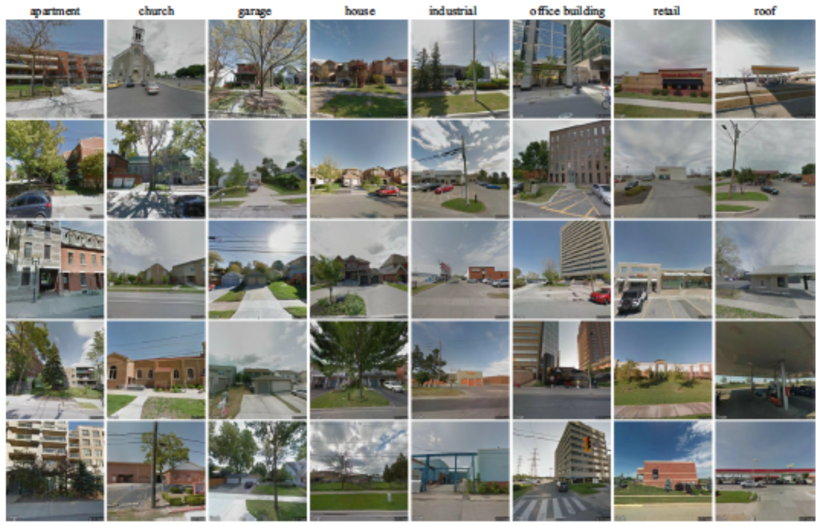


Fig. 7. Number of street view images of each building class.



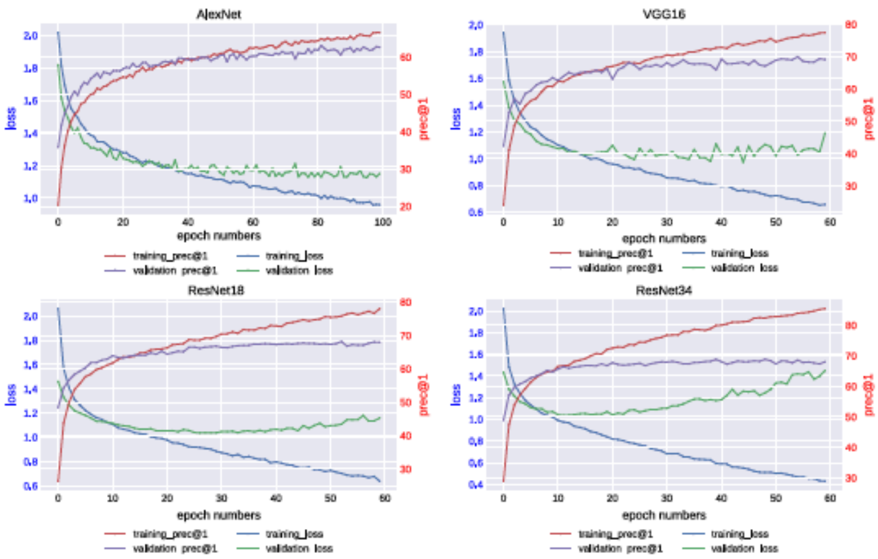


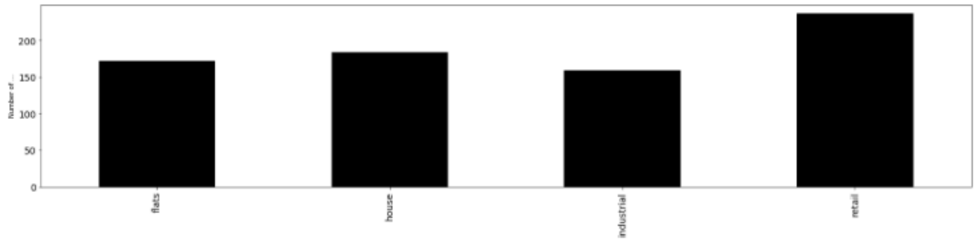
Fig. 10. The learning and top 1-precision curves of the four networks, i.e. AlexNet (Top-left), VGG16 (Top-right), ResNet18 (Bottom-left) and ResNet34 (Bottom-right). It can be seen that training losses of the four networks reduce as the epochs increase. Besides, the validation learning curve of AlexNet converges until 80 epochs, and those of the other three networks can converge within 60 epochs. Overfitting behaviors are found in ResNet18 and ResNet34, and it is more severe in ResNet34. One plausible reason is that the total parameter number of ResNet34 (21 million) is more than that of ResNet18 (11 million). As shown by top-1 precisions, AlexNet can achieve about 65%, while the other networks can obtain about 70%.

Fig 6.49 – *Kang et Al Comparison: Image Instance Count, Class Samples and Model Overfit()*

Quick Inspection of Results

Ground Truth Property Type Distribution

```
In [56]: #Check Distribution of Property Types FOR GROUND TRUTH
         plotdat(df_ldd_label_join,'C_Truth')
```



CNN PREDICTION Property Type Distribution

```
In [57]: #Check Distribution of Property Types FOR CLASSFIER
         plotdat(df_ldd_label_join,'Predicted_Label')
```

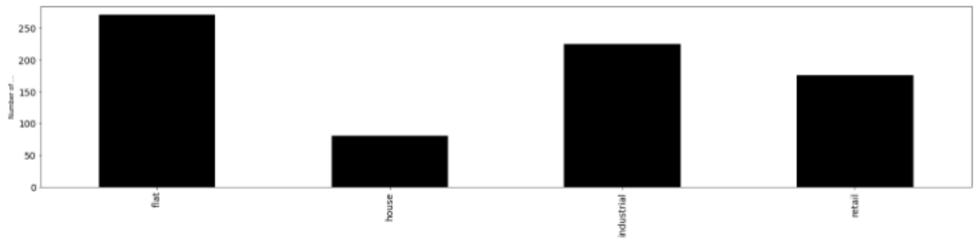


Fig 6.51 *– Ground Truth vs CNN Prediction Scores for an N=250 Type=House, Flat, Office, Retail, Industrial Label Set Training Run*

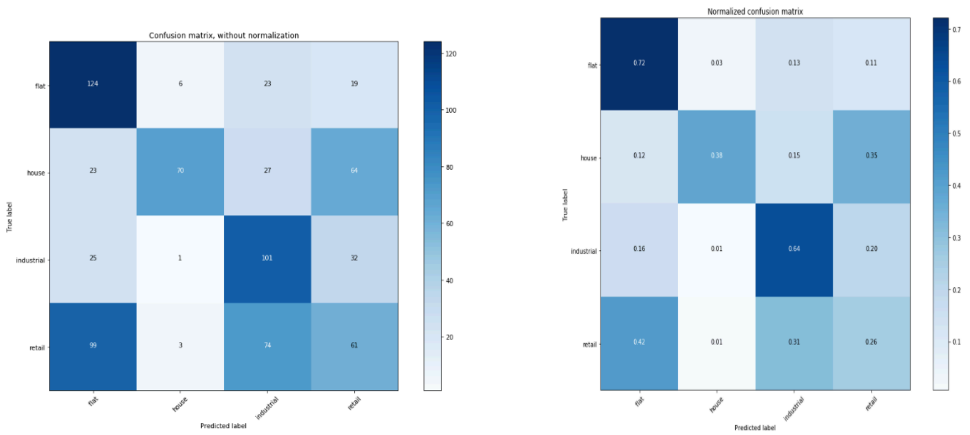


Fig 6.52 *– Confusion Matrix for an N=250 Type=House, Flat, Office, Retail, Industrial Label Set Training Run*

6.4.2 Prediction Anomaly Grid and Gallery

We have collated a table of classified anomalies, which identify the lack of distinction in many London Building Types, which can confound a machine learner. They also reflect on the dense and proliferated presence of mixed use property and developments. Figures 7.8 and 7.8 b provides spatial context of 2 further classification anomaly examples.

Locale	Image ID	AB Code	Prediction	Confidence	X	Y
Brent	**4747**	RD03	office	0.98709	51.540141	-0.265168
Brent	**4891**	CI04	retail	0.99605	51.535536	-0.276706
Brent	**4845**	RD06	retail	0.99768	51.537491	-0.193228
Brent	**4922**	RD06	office	0.91671	51.530025	-0.21538
Brent	**6548**	RD03	industrial	0.46663	51.557643	-0.309094
Brent	**4836**	CI	retail	0.77995	51.568356	-0.235898
Brent	**4873**	RD04	retail	0.54531	51.5335	-0.235257
Croydon	**14957**	CI	office	0.99271	51.353995	-0.119483
Croydon	**12815**	RD02	office	0.99271	51.309968	-0.098999
Croydon	**14792**	RD06	industrial	0.96298	51.407773	-0.101172
Croydon	**14073**	CI01MG	house	0.99804	51.356772	-0.120446
Croydon	**14096**	RD02	office	0.99271	51.344276	-0.122408
Croydon	**12508**	RD06	retail	0.55372	51.357076	-0.105869
Croydon	**13801**	RD02	flat	0.60564	51.320144	-0.060905

Fig 6.53 *– Image Anomaly Grid*

Image 4747

Predicted as office, the rigid angular and muted coloring of the façade coupled with the direct but opaque assemblage interface might suggest this flat complex has office like attributes.

Image 4836

Predicted as Retail, this is in actual fact an Industrial Property type. The For Sale signage and light coloring of the pedestrian/car set back might be factors in the classifier returning its prediction.

Image 6548

Address base identifies this property as semi detached, the prediction returned it as Industrial. The large open and concreted floor space and stark setting might have been factors in this classification.

Image 4922

AddressBase identifies this property as a flat, whilst the classifier predicts it as Office. The urban setting and the scaffolding may be factors in this result.

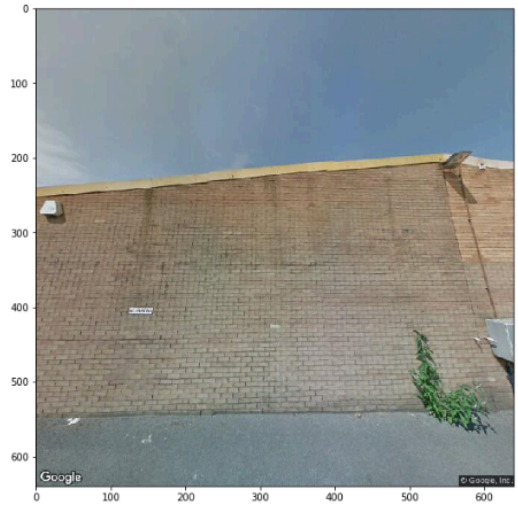


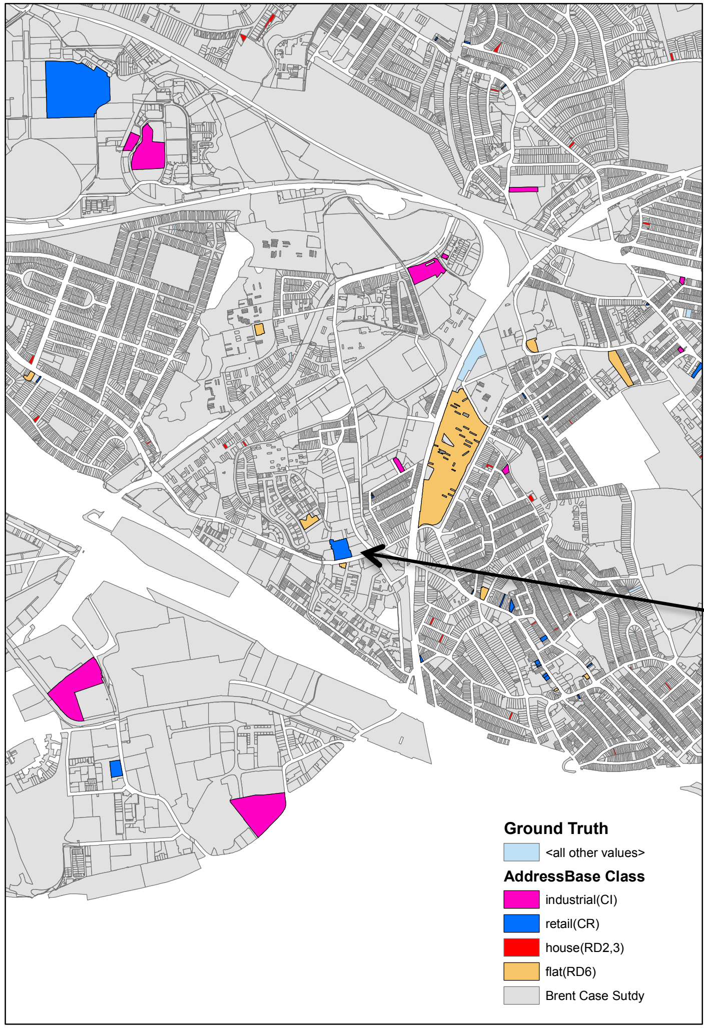
Image 14073

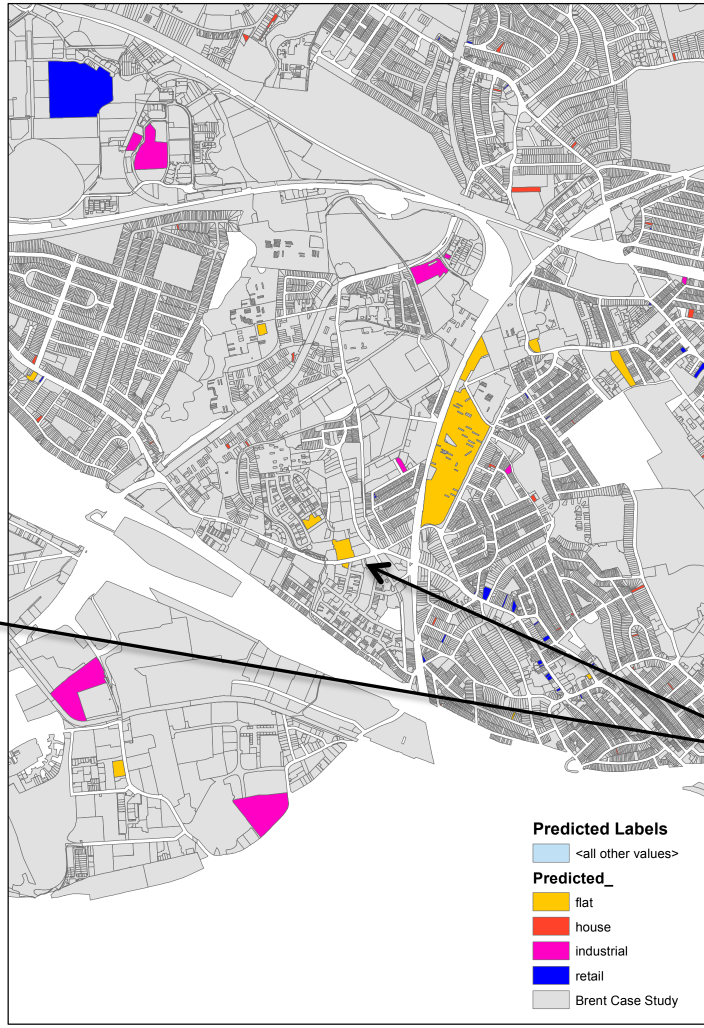
This Croydon property is an Industrial unit. The poor accessibility afforded the Google Street View Van, has produced an inadequate image of the property. The classifier predicts this as house, which might reflect the panoramic positioning of the photo which is similar to most of the suburban images of houses, detached and semi detached. Ideally this would have been picked up by the pre classification process in a more stringent processing stage.

Image 4845

AddressBase marks this as a house, whilst this is predicted as Retail. Clearly in the image we have both, but the Classifier has probably been swayed by the signage and by the filter which omits new development with less than 4 units.

***Fig 6.55** – Ground Truth vs Predicted Type - Building Footprints and Submitted Planning Application Proposal - Bromley*





Erection of part three-, five- and six-storey building to provide health centre, community centre, radio station, cafe, retail shop, 25 one-bedroom flats, 32 two-bedroom flats and 2 studio flat, formation of 47 basement car-parking spaces (of which (cont)...

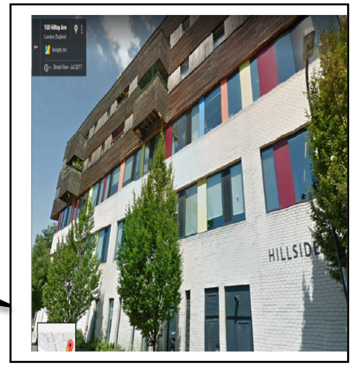
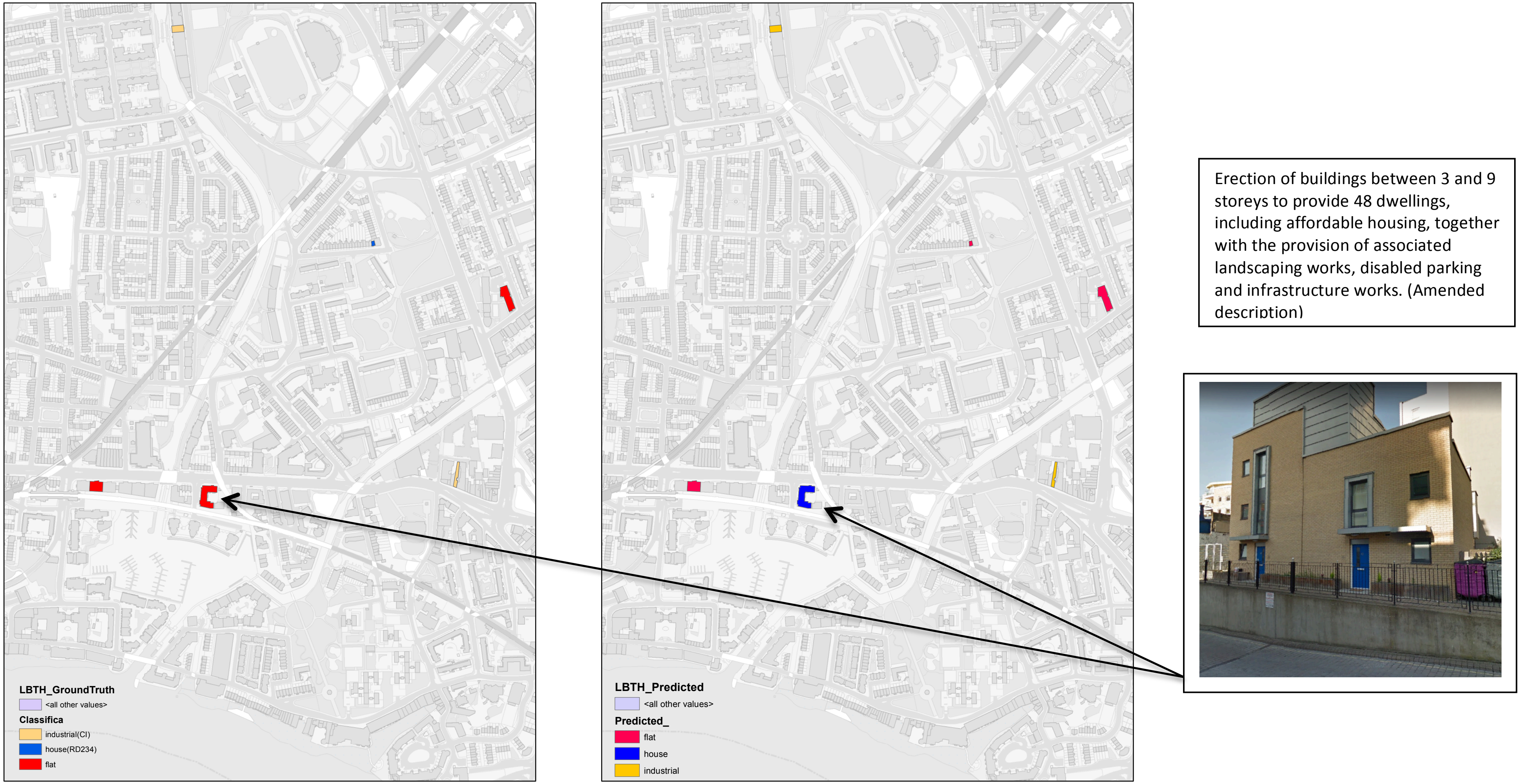
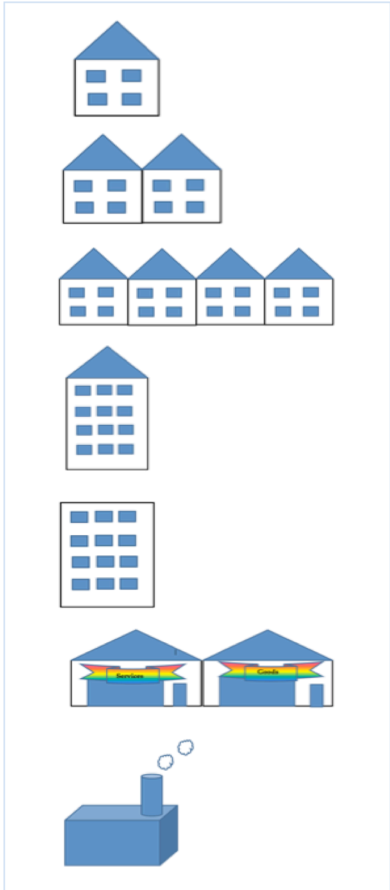


***Fig 6.25** – Ground Truth vs Predicted Type - Building Footprints and Submitted Planning Application Proposal – Tower Hamlets*





< Detached

< Semi- Detached

< Terraces

< Apartments

< Office

< Retail

<Industrial

Fig 6.56 – *Idealized Building Form Classes.*

Fig 6.26 (overleaf) highlights overlapping typological areas which generated adversarial image instances for our machine learning model.

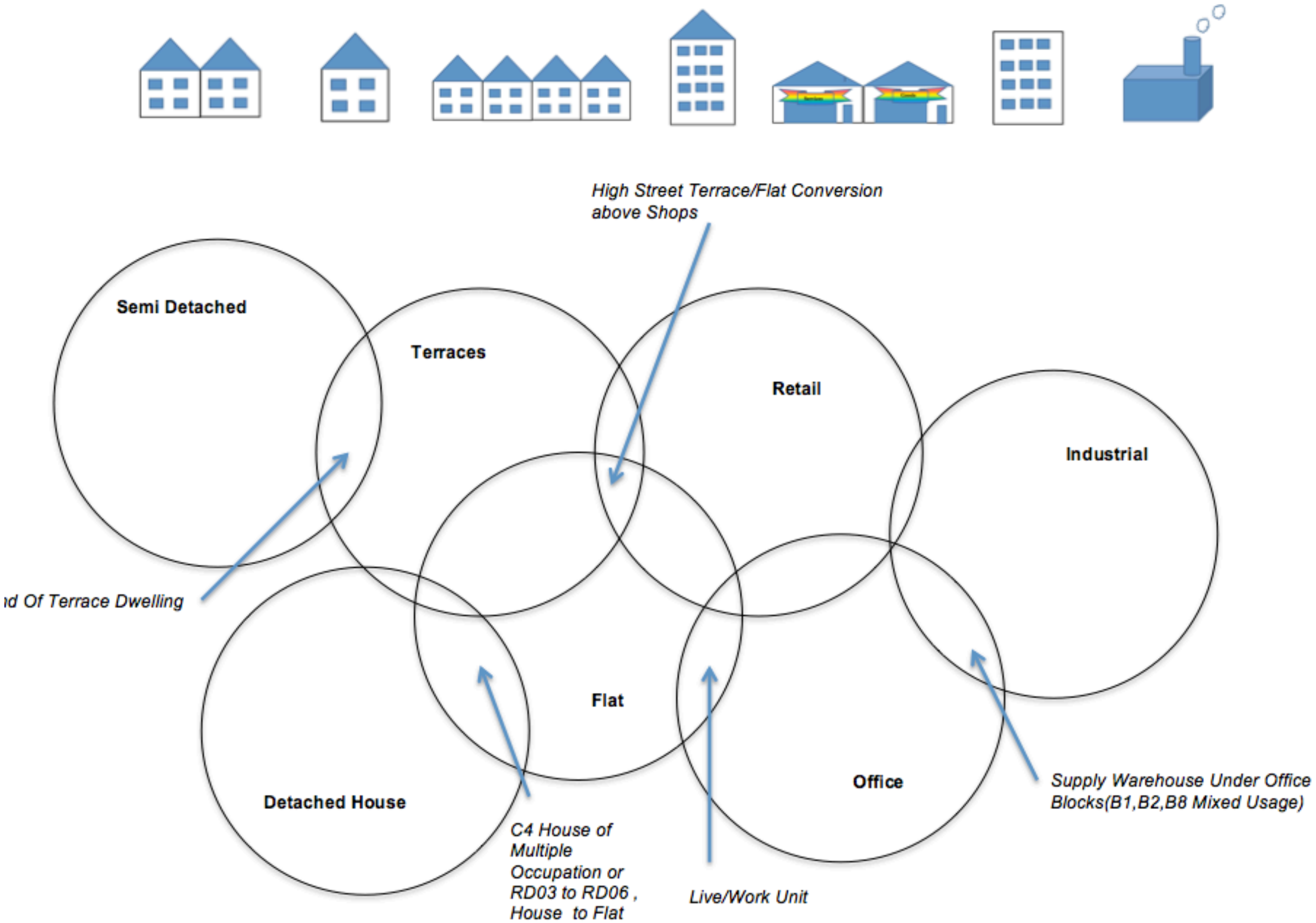


Fig 6.57 – *Mixed Used Development and Building Type Cross Pollination highlighting the major issue faced by our Image Classification Task: large numbers of mixed Building Types and building form and function ambiguity in the London New Development dataset.*

6.4.3 Classification Results

Despite revisiting the Workflow Design(both in terms of more rigid Occ and Anomaly removal, and in applying several label set creation variati strategies) and methodically applying the most common Machine Lear Techniques widely used for treating model overfit(freezing layers, data augmentation, weight regularization and dropout)we found there to be significant improvement in the problem of model overfit.

Having the Kang et Al Us Building Instance data was a useful referenc achieved approx. 70% accuracy rates but with upwards of 1000 instan class.

Given that the LDD dataset yields nearly 70000 instances of new development, it is surprising that after pre processing and set creation filtering, we were unable to gather samples from our target categories i similar and sufficient supply.

Furthermore we found that training our model on the Kang et Al releas produced similarly disappointing rates of accuracy on the London Data suggest the use of an alternative building rejection class, which we ha results with, again down to the data quantity/quality.

This suggests that whilst the overwhelming issue here is a lack of data also infer that building instance CNN models trained on Street Level Im do not necessarily generalize well. That is to say, Street Scene Image Datasets must also be considered in terms of their geographically loca features and content. Furthermore a successful machine learning work based on Street Level Imagery of this kind would most likely involve us hybrid and ensemble method, blending different datasets and configuri model element/stage toward the type of data in use at a particular poin workflow.

Taking these facts into consideration as a whole, we can consequently more confident in reaching a conclusion into what this says about the r of London New Build Development in comparison to US Cities but also regard to how this allows us to identify predominant characteristics of New Build as identified in the London Development Database.

Ultimately this confirms the overwhelming presence of mixed use development in a highly dense setting(accounted for by the prevalence occluded images) and which features a significant number of incomple permissions, non major scale conversions. To a lesser extent the preva of anomalous images created by a large number of inaccessible sites i suggested from these results. These points are considered and their significance in urban science and planning terms are dealt with in furth detail in the study conclusion.

6.4.3 Classification Results Summary

Figure 6.19 documents the full set of operational problems faced in the study workflow and the mitigations taken for each issue.

Figure 6.11 highlights the difficulty and trade offs applied to the overall balance of the workflow when these steps were taken.

Figure 6.23 and Figure 7.8 highlights examples of mixed use building category cross over in the London New Build DataSet in a Set diagram format.

We summarize our findings as follows:

1 - Insufficient Numbers of Data and Data Class Representation

After the Pre processing stages the Master Model of upwards of 80k Images returned a disappointing size of training data both in terms of overall quantity and in providing representation of the various building types.

2 - ImageNet DataSet trained weights do not transfer well onto our Urban Street Scene based target Dataset

Building Object Types possibly not adequately represented in the ImageNet SynSet Hierarchical Structure. Building Objects are in the minority and fall under the artifcact synset node.

Recommended Action 1: Use Weights from Models trained on Sun or Places 365 data sets.
Result: Not Attempted

Recommended Action 2: Fine-tune the weights of the pretrained network by continuing the backpropagation.
Result: Small Improvement in Accuracy Rate

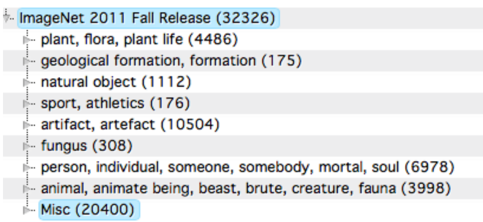


Fig 6.58 – *ImageNet Synset Nodes*

3 - Prevalence of Mixed Use Building Typologies vs Lack of Distinct Building Types in the LDD Data Set

Recommended Action 1: Apply the US Cities Trained Weights as Classifier on the LDD Dataset however the CNN will struggle to differentiate some of the categories identified regardless of the contingency option adopted.
Result:

4 – Workflow Design that depends on Occlusion Removal requires a large amount of data to be discarded.

5 – Fuzzy Control Filtering cannot mitigate for invalid data (e.g. unrefined or outdated geo referencing in a rapidly changing urban environment)

6- Applying some method of varying Field of View of Images Selection would be beneficial however, as identified by the ConvNet Visualization sub-stage and recent research into how CNNs reach their decisions, current CNN models are able to correctly Label images with multiple candidate objects if supplied with sufficient quantity of image training data.

Whilst we achieved initial accuracy rates of >70% this was accompanied by considerable levels of model overfitting. Returning our attention to the Pre processing stage we identified that discriminate use of the > 1000 fine grained labels of the Places 365 as occlusion filters was difficult and allowed for a small continued presence of occluded images and highlights how even a small number of outliers can confuse our CNN model. Applying commonly used CNN techniques for model fine tuning were found to have a improvement however this was small and highlighted how these techniques were more suitable for application during the latter stages of a successful Classification workflow.

Given the above and taking the results obtained from the study's benchmark training upon the Kang et Al released Data Model it was apparent that applying a relatively simple object classification task on a London Based Street Scene Image dataset is surprisingly problematic and highlights the visually noisy and highly dense nature of London new Building development that is also typified by high prevalence of inaccessible public/private assemblage street to property interfaces.