

Quantifying On-Street Parking

Tamsin Forbes
*Aviation, International & Maritime
Statistics
Department for Transport
London, UK
tamsin.forbes@dft.gov.uk*

Abstract—In this report I present a methodology to iteratively improve upon an estimate of on-street parking capacity in a residential test area. I combine information from multiple flat file and geospatial vector and raster datasets including aerial photography. As well as using various standard data wrangling and geospatial techniques I present an application of the isoperimetric quotient to identifying road junctions and apply the bag of visual words model to identify zebra crossings from aerial imagery. This is intended to inform electric vehicle infrastructure planning in residential areas.

Keywords—electric vehicle, chargepoint, parking, geospatial, raster, bag of visual words, image processing, isoperimetric, machine vision

I. INTRODUCTION

A. Motivation

The Office for Zero Emission Vehicles (OZEV) with the Environment Statistics team at the Department for Transport (DfT) are currently planning the future provision across the UK of electric vehicle (EV) chargepoints, as part of the commitment to reducing emissions by 2050 [1]. We imagine a future where all vehicles are fully or partially electric, and every vehicle requires adequate access to a chargepoint. Charging an EV can take considerably longer than refuelling a combustion engine. Different chargepoint types (slow, fast, rapid) can charge an EV partly or fully at various speeds, depending on power rating, ranging from five minutes to ten hours. The demand on the electricity grid is an important consideration, and to mitigate this the majority of charging should take place at night, outside peak hours of electricity consumption [1]. Thus, it is anticipated that there will be many chargepoints located on, or nearby the property of the EV owner. Hence it is important to estimate parking availability. It is expected that properties will fall into one of the following three categories,

1. Off-street parking: the property has a driveway or garage.
2. Adequate on-street parking: there is enough room for all or most vehicles to park within some specified short distance of the property.
3. Inadequate on-street parking: there is not enough room for all or most vehicles to park within some specified short distance of the property.

There is no definitive existing dataset that quantifies residential parking availability for the UK. There are various datasets which may include some information. For example, HM Land registry and estate agency collections, such as Zoopla, may contain garage or off-street parking details for a property. Data on on-street parking restrictions is usually held

by Local Authorities in various formats and there is no centralised standardised dataset.

B. Objectives

The aim of this work is to inform infrastructure planning of public EV chargepoints by estimating the proportion of on-street parking. I restrict the scope to a one-kilometre square residential test area in the borough of Ealing.

C. Solution overview

My solution is to combine multiple datasets to exclude areas where on-street parking is not possible or strictly prohibited, for example, due to road features such as bus lanes, road junctions and driveways. Subtracting these areas from the total residential road length or road section shapes will leave only road sections or lengths where on-street parking is possible. I ignore parking restrictions due to permissions and focus on restrictions due to safety or flow of traffic. For example, driveways and areas near bus stops must remain unobstructed at all times. However, residential parking permits are controlled by Local Authorities, and it is within their remit to change the conditions of parking hence they could decide to install public (or resident only) EV chargepoints in these spaces.

D. Main findings & relevance

This project demonstrates the viability of this method and further contributes to the knowledge base for EV chargepoint infrastructure planning. Combining multiple datasets shows the contribution of each and provides an indication of the breadth and complexity of the task to extend to the UK. It also demonstrates which datasets provide the quick wins. For example, an estimate that ignored the image data may fail to exclude parking on zebra crossings however the modal would be simpler and not significantly impact the estimate given the relative sparsity of zebra crossings.

This project has laid the groundwork such that the team at DfT will be able to continue the work and extend to the UK. There is also the possibility of a cross government collaboration to produce an online tool that will allow Local Authorities to assess a given area for on and off-street parking potential. This information will enable Local Authorities to plan appropriately for the deployment of EV chargepoints.

II. BACKGROUND

A. Geospatial analysis

Much of the data is geospatial in nature comprising polygons, polylines or points within the British National Grid (BNG) coordinate reference system together with other features relating to each shape. Each observation in these datasets is a geometric shape with a unique identifier and other fields containing attributes of that geometric feature. Each shape corresponds to its physical location and other properties such as area, perimeter, length and angle are calculable. Geospatial analysis techniques such as spatial joins can be used to filter and combine these data based on proximity to residential buildings, or adjacency to neighbouring shapes with particular properties. Buffering can be used to determine the area of influence of a shape by extending its perimeter a given distance.

B. Isoperimetric quotient

One of the main challenges of this project is to identify road junctions. This is important because parking near road junctions may be restricted and the differences in parking potential between a crossroads and a T-junction. Road junctions are not specifically defined in the datasets so I resort to using their spatial attributes to aid identification. The isoperimetric quotient is essentially a measure of compactness of a shape. The related isoperimetric inequality states that square of the perimeter of a shape is greater than or equal to its area times 4π , or

$$L^2 \geq 4\pi A \quad (1)$$

where L is the length of the perimeter of the shape and A is its area. In two-dimensional space equality holds only when the shape is a circle. The isoperimetric quotient, Q , of a shape is defined as the ratio of its area and that of a circle having the same perimeter [2], and is given by

$$Q = \frac{4\pi A}{L^2} \quad (2)$$

Combining with (1) implies that $Q \leq 1$, hence Q may also be thought of as indicating how close to a circle a shape is. This is useful to distinguish road junctions, which are often squarish, from longer stretches of road.

C. Machine Vision

The Bag of Visual Words (BoVW) machine vision model has perhaps now been superseded by deep learning convolutional neural net (CNN) approaches, however, there are advantages to BoVW. It is relatively straightforward, transparent, and interpretable. In a binary classification scenario, a set of training images of the positive image (in this case aerial images of zebra crossings) is used to collect keypoints. Keypoints are identified by the change in pixel intensity values and then described using information from the surrounding pixels to contextualise and create a feature vector. The number of keypoints located is dependent on number of pixels present and the level of thresholding applied, but can be very large, and restricting this is important to reducing computational expenditure. The feature vectors

are clustered to create the visual vocabulary; essentially a set of vectors that represent the different types of keypoint. All images are then processed to find keypoints and compare to the visual vocabulary vectors, producing a histogram of counts of each visual word for each image. Each image is now represented by a feature histogram and these features are used in a classic machine learning model such as support vector machine (SVM) binary classifier. Hence, the BoVW model provides a method to extract rich descriptive features from images in a standard way.

D. CNN to Map Houses Suitable for EV Home Charging

This paper presents a novel approach to detecting residential homes with potential to install private EV chargepoints. Using open source Google street view images in the vicinity of residential homes point location and a combination of transfer learning and CNN to detect the presence of driveways in the images [3]. I considered attempting a similar approach using aerial photography, however, the overheads for setting up a deep learning CNN model require a Graphics Processing Unit (GPU), and proved beyond the scope and timescale of this project. Theoretically, this is a more scalable approach than BoVW.

E. EMU Analytics

Emu Analytics has developed a product using the Land Registry INSPIRE Index (LRII) polygons, which show the extent of a property, and single structure outlines from Ordnance Survey Open Map (OSOM). The distance from the property frontage to the building is used to calculate a likelihood that the property can support off-street parking [4] and the subtraction of this information is used to infer the likelihood that a property will require on-street parking. As a baseline of potential parking demand this product has some merit, however, they are not able to restrict the data entirely to residential properties, nor delineate all connected properties, such as terraces. More pertinently it is not free, and as a public sector enterprise, DfT has access to free and superior data sources under the Public Sector Geospatial Agreement (PSGA).

F. Zebra Crossing Detection for the Partially Sighted

This paper discusses methods of identifying zebra crossing from street view road scenes with the aim of developing a tool to assist the partially sighted. In a street view road scene the actually parallel lines of a zebra crossing appear non-parallel and when extended further may intersect at the vanishing point. Searching for the vanishing point of concurrent lines detected through edge detection appears to be a relatively deterministic approach, having ascertained appropriate thresholds. Accuracy is subject to detection of enough candidate lines to test for coincidence [5]. For aerial zebra crossing images I considered there may be a correspondingly similar way to detect groups of parallel edges, and found so.

G. Zebra Crossing Detection from Aerial Imagery

This method uses Histogram of Oriented Gradients (HOG) and Local Binary Pattern Histogram (LBPH) features to describe zebra crossings in aerial imagery, using SVM model for classification [6]. The motivation is to improve existing open-source geospatial datasets such as *OpenStreetMap* to provide better information for the visually impaired. This approach achieved excellent results in testing, for example over 97% in precision, recall and accuracy for the LBPH and SVM with Radial Basis Function (RBF).

III. MATERIALS AND METHODS

This project combines multiple datasets many of which are Ordnance Survey Master Maps (OSMM) geospatial layers and freely accessible to public sector employees via the Public Sector Geospatial Agreement (PSGA).

A. AddressBase Plus



Fig. 1. AddressBase Plus example residential property point locations.

AddressBase Plus is an OSMM product which contains a unique property reference number (UPRN) for every postal address in the UK, along with point geometry for its location and the Topographic Identifier (TOID) of the building footprint in the Topographic Area layer the UK. Each address has an address type, for example, residential dwelling, commercial, etc. The classification scheme contains over 500 address types organized in a hierarchy, allowing detailed filtering of property types [7]. No personal information is included since the addresses are not linked to individuals.

B. Topographic Area Layer

OSMM Topographic Area Layer comprises a jigsaw of vector polygons making up the entire surface of the UK. It includes various descriptive fields such as buildings, roads, land etc. Every shape has a unique TOID and only intersects which other shapes on adjacent edges; there is no overlap of polygon area. This is useful to identify roads adjacent and nearby to residential properties.

C. Highways Layer

OSMM Highways Layer represents the midline of roads as polylines (links) and intersections between these as points



Fig. 2. Topographic Area Layer example showing buildings (brown), land (green), paths (grey), roads (cyan), roadside edge (magenta).

(nodes). Length of road is therefore easily calculable and other useful attributes are included such average road width and minimum road width. The length and widths of roads is used to determine how many parking spaces the length of the road can accommodate and if the road is wide enough to allow parking on one side, both sides or neither. There is not a full one-one correspondence between roads in the Highways Layer roads and Topographic Area Layer as the former includes ‘roads’ that might be untarmacked and more akin to tracks used for access. Therefore, the Highways Layer nodes do not correspond directly to road junctions. It is important to accurately identify junctions and different types of junction as it is not permissible to park close to a crossroads, but it may be possible to park on parts of a T-junction.

D. National Public Transport Access Nodes (NaPTAN)

The NaPTAN dataset contains point location data for many public transport access points such as railway stations, airports, and bus stops. Unfortunately, there is some inaccuracy in these data most likely due to longitude and latitude recorded at insufficient precision. Some of the bus stop locations are given a little way from the actual bus stop, hence these data are pre-processed to snap the bus stop locations to the nearest road. See Fig. 3 for example.



Fig. 3. Highways layers links (magenta lines), nodes (cyan points), NaPTAN bus stops (red points), Topographic Area Layer roads (green), on APGB 25cm resolution background.



Fig. 4. APGB RGB 25 cm showing zebra crossing, with Topography Area Layer road polygon edges (green) and Highways Layer road links (magenta) and nodes (cyan).

E. Aerial Photography Great Britain (APGB)

APGB products are raster images of one-kilometre square grids that align with the British National Grid (BNG) system. They combine multiple aerial photographs taken between 1 April and 31 October, usually over one year, but can be multiple years. The RGB 25 centimetre renders 1 pixel length for every 25 centimetre of ground level distance and contains values in the range [0, 255] for three channels of the visible spectrum, Red, Green and Blue [8]. The resolution in these images is sufficient to clearly see zebra crossings and other permanent no parking road markings as can been seen in Fig. 4. This will be used in the machine vision stage of the methodology. These images are already processed to remove any personal information such as people, in accordance with GDPR.

F. Off-street parking outputs

The outputs from the Off-street parking project comprise a dataset for the same test area (BNG TQ1980) of the UPRNs of properties with off-street parking possible in the front garden. This was validated against a manually labelled dataset. From these it is possible to derive the driveway width by considering the intersection between the roadside edge and the manmade surface area of the front garden, see Fig 5. Alternatively, an average driveway width could be used and applied to each property labelled as having off-street parking.



Fig. 5. Off-street parking outputs showing front garden manmade surface (magenta) intersection with roadside edge (blue).

G. Data pre-processing

The NaPTAN dataset is a little inaccurate and needs to be snapped to the nearest road section to better represent the actual location of each bus stop on the road.

The Topography Area layer is filtered to just roads and tracks, and further filtered to those roads that are within short walking distance of residential properties using spatial.

The Highways layer is filtered to just the road links and road nodes and can then be further reduced using the road widths features. I used QGIS open-source geospatial software to clip the large polygon geopackage files to the one-kilometre square grid test area. The APGB files are then clipped to just the roads to reduce the number of pixels for the machine vision algorithm to process.

H. Methodology

The basic approach is to use the simple formula

$$N = \frac{2L}{P} \quad (2)$$

where N is the number of parking spaces, L is the road length and P is the parking space length. P is assumed to be 5 metres to accommodate the standard length of a parking space in the UK, 4.8 metres, plus some wiggle room for parallel parking. The formula assumes parking is possible on both sides of the road, hence the length of the road is multiplied by a factor of two. We attempt to iteratively improve upon an estimate of on-street parking capacity by subtracting sections of road where it is strictly not permitted to park for various reasons listed here;

- 1) Elimination of road types that do not permit parking such as motorways and dual carriageways, using Highways layer road type field.
- 2) Determine where parking is possible on one, both or neither side of the road using road width features, taking average car width at 2.4 metres (standard UK parking space width).
- 3) Identify road junctions using isoperimetric quotient and eliminate parking nearby these areas.
- 4) Remove 5 metres either side of bus stop locations.
- 5) Remove inferred driveway widths, or average driveway widths given by off-street parking results.
- 6) Identify strict no parking areas from permanent road markings such as zebra crossings using machine vision techniques.
- 7) Remove remaining isolated sections less than 5 metres in length which are now too short to accommodate parking.

These exclusions result in a set of road sections (polygons), or boundaries of road sections (polylines) where parking is still possible and to which formula (1) or its derivatives can be applied. The subtractions of most relevance, due to complexity, are 3 and 6, hence the remainder of this report will focus on these.

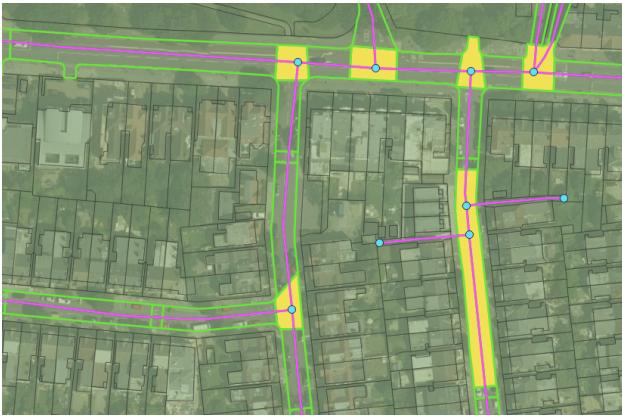


Fig. 6. Highlighted yellow sections of Topography Area layer roads (green border) intersection with Highways Node layer (cyan points). Highways Link layer shown for reference (magenta lines).

I. Identifying road junctions methodology

It is important to identify road junctions since parking restriction in their vicinity differ by type. It is possible and permissible, if the road is wide enough, to park across the T bar of a T-junction. Unfortunately, the various layers do not precisely correspond in their attributes. The Highways layers include small, usually unsurfaced, tracks in their road classifications, whereas the Topographic Area layers do not. These might be gravel tracks to rear garages or sheds, for example. Hence, there are more nodes, representing the junctions between any road or track, in the Highways Layers than in the Topographic Area layers. Fig 6 illustrates this showing a yellow highlighted long road section intersecting with two nodes, as there are two tracks spurring off it. The Topographic Area layer treats this as one polygon, rather than four if there were perfect correspondence between the two. Since most junction road sections are fairly compact in shape, the isoperimetric quotient may be used to discriminate between the true road junctions and those with tracks.

J. Identifying zebra crossings methodology

Of the strict no parking permanent road markings zebra crossings are the most distinct, and for their size inflict parking restrictions on a relatively large area. This is due to the necessity of maintaining line of sight of where people may be crossing from either side of the road at a great enough distance for drivers to notice and decelerate safely to a stop.

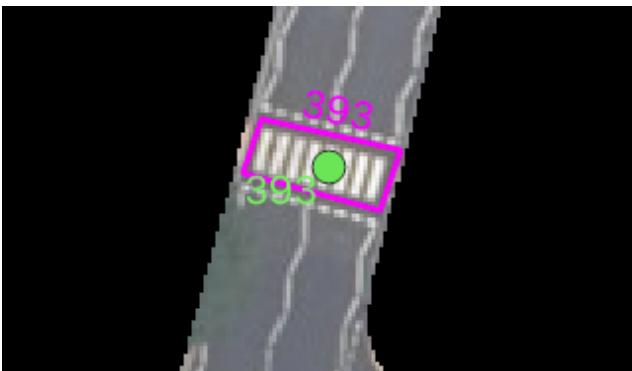


Fig. 7. Example of one of the *Letter Set* zebra crossing polygons (magenta). Corresponding Point dataset (green) also collected in QGIS. Aerial image has been trimmed to roads only.

To collect keypoints only from images of zebra crossings I used open-source geospatial software to manually draw close fitting polygons around images of zebra crossings in the aerial photography of my test area, and combined these into a new geopackage file, see Fig 7. I refer to this as the *Letter Set* since all subsequent visual vocab is built from it. Since the original one-kilometre square test grid (BNG TQ1980) does not contain sufficient zebra crossing examples, I expanded my search to include all 82 one-kilometre square grids available to me for the Borough of Ealing. I encountered some anomalies such as private land zebra crossings that are narrower and also used to demarcate pavement where a raised curb is not enforced, hence they are not always perpendicular to a road. There exists some occlusion in the aerial images, for example, by trees or vehicles. Where occlusion is extreme, such as at least 50% coverage, I marked these samples to be excluded from the *Letter Set*, since to extract a useful set of visual words that describe images of zebra crossings it is important to use clean images.

To prepare the training and test sets I created a subset of the roads containing zebra crossings and for each road link interpolated points every two metres along the road. I use these points to define the midpoint of 8 by 8 metre bounding boxes, hence creating a set of polygons that follow the midline of the road. I then cut these polygons out of the 25 centimetre APGB images creating a set of individual images that together define a sliding window path along the roads, I refer to these as *patches*, see Fig 8. The images patches are 32 by 32 pixels.



Fig. 8. Five image patches at two metre intervals along a road showing progress over a zebra crossing moving west to east.

To label each patch I overlaid the *Letter Set* polygons with the patches and labelled a patch as containing a zebra crossing (positive class) if the areas intersect. In retrospect this was too naïve, as a few patches containing very little intersection with a zebra crossing were thus labelled as positive. To improve this in future I can set a minimum area overlap to indicate positive. I then split each image patch into a training or test set, ensuring a balance of the two classes and a ratio of 70:30 of training images to test images. The total number of images in the test set is 196, with 98 in each class.



Fig. 9. Example of SIFT keypoints ringed in a zebra crossing from the *Letter Set*. The images are small, ~600 pixels, hence not many keypoints.

I used Scale Invariant Feature Transform (SIFT) features to extract keypoints from the *Letter Set* (see Fig 9) and clustered these into different vocab sizes. For each vocab set and training and test patch I created a histogram of features describing the frequency distribution of each visual word from the vocab within each patch. Hence, each patch image is described by a single numerical vector of the same length, which can be easily compared for similarity to other image descriptors using standard distance measures such as the Euclidian distance. During experimentation I discovered that due to the small size of the *Letter Set* images, on average about 600 pixels, some images did not return any keypoints. Due to time constraints, I did not explore further image processing to create clearer images and ensure many keypoints were found. In future I could use another SIFT algorithm such as Dense SIFT which returns a spread of keypoints extracted evenly across the image. Given the visually distinct and contrasting nature of zebra crossings my expectation was that this method should work well regardless. It may be advantageous to use only strong keypoints, such as those I collected from the *Letter Set* using SIFT since this might avoid confusion with other road markings similar to weak images of zebra crossings. However, this would require further testing.

Using the *Letter Set* I created a range of vocab sets with vocab sizes 5, 10, 20, 50, 100 and 250. In each case the *Letter Set* feature vectors were clustered using k-means into the number of clusters indicated by the vocab size. The centroid of each cluster became the representative feature vector for that visual word. For each vocab size I computed feature histograms for the training and test sets and then classified the images using SVM supervised binary classifier.

IV. ANALYSES AND RESULTS

A. Identifying road junctions analysis and results

I completed some brief experiments using compactness to distinguish road junctions. After filtering the road sections in the Topographic Area layer that intersect with nodes from the Highways Node layer I calculated the area and perimeter of each road section and from these computed the isoperimetric quotient to obtain a compactness score. Fig 10 shows the distribution of compactness for this set. As expected, many road sections intersecting with nodes are substantially more

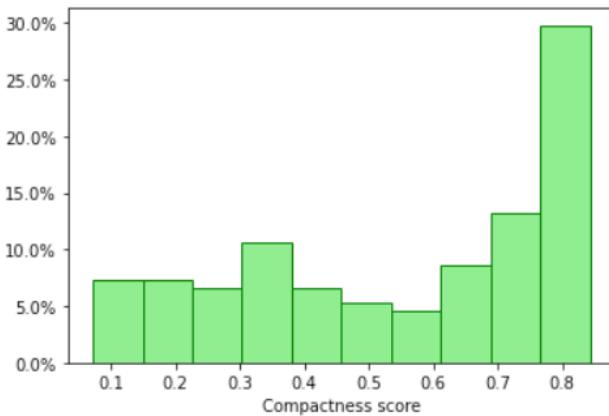


Fig. 10. Histogram of compactness score of all road sections in Topographic Area layer that intersect with a node from the Highways node layer. In 2D a circle has perfect compactness with a score of 1.



Fig. 11. Road sections intersecting nodes split by compactness threshold of 0.6. Yellow has compactness ≥ 0.6 and coral has compactness < 0.6 . The crossroad in the top right has a low compactness score due to its concave shape, rather than its length.

compact, with scores over 0.7, indicating correspondence between nodes and true road junctions. However, a significant number have comparatively low compactness indicating these are actually longer stretches of road.

I experimented with various compactness thresholds, plotting the road sections coloured by which side of the threshold the compactness score lay. Fig 11 shows an example with a threshold of 0.6. While the long road sections with low compactness are identified (left hand side) this plot illustrates a problem with the methodology regarding crossroads where the shape includes parts of the road leading into the crossroad. This results in a cross shaped road junction, which is somewhat concave and thus has a lower compactness score. Further work is required to find a robust way to identify road junctions. I considered calculating the compactness score from the smallest bounding convex shape for each road section, however, did not have time to fully explore this idea. Another possibility is to use the Highways layer links and nodes as a network representation. Calculating the node degree may be sufficient to indicate the type of junction, for example node degree 3 implies a T-junction, whereas node degree 4 implies a crossroad. However, there would be complications with this method too due to road links splitting to navigate road furniture, and potentially labelling phantom T-junctions. The top left road spur in Fig 4 demonstrates an example of road furniture splitting the road link.

B. Identifying zebra crossings analysis and results

To ensure robustness I used five-fold cross validation and set up a hyperparameter grid search to test multiple SVM models. I tried four different kernels; linear, sigmoid, radial basis function (RBF), and polynomial. I varied the

parameter	set of values
C	[0.1, 1.0, 10.0, 100.0]
gamma	[0.01, 0.1, 1.0, 10.0]
kernel	['linear', 'rbf', 'poly', 'sigmoid']
degree	[2, 3, 4]

Fig. 12. Table of sets of hyperparameters. Linear kernel only uses *C* and only the polynomial kernel uses *degree*.

vocab_size	best_params	accuracy
5	{'C': 1.0, 'degree': 2, 'gamma': 1.0, 'kernel': 'rbf'}	0.780612
10	{'C': 1.0, 'degree': 3, 'gamma': 1.0, 'kernel': 'poly'}	0.780612
20	{'C': 10.0, 'degree': 2, 'gamma': 0.1, 'kernel': 'rbf'}	0.770408
50	{'C': 10.0, 'degree': 2, 'gamma': 1.0, 'kernel': 'rbf'}	0.790816
100	{'C': 1.0, 'degree': 3, 'gamma': 1.0, 'kernel': 'poly'}	0.821429
250	{'C': 10.0, 'degree': 2, 'gamma': 1.0, 'kernel': 'rbf'}	0.806122

Fig. 13. Table of model parameters and accuracy scores for the best performing models for each vocab size. For RBF kernel ignore degree.

regularisation constant C , γ and degree. The linear kernel only depends on C . The non-linear kernels all depend on C and γ , but only the polynomial kernel uses degree, and this is ignored by the other kernels. ‘Low’ C values encourage a larger margin in the model, hence a simpler decision function, at the cost of training accuracy. However, a larger margin helps separate the classes more smoothly, moderating overfitting. The γ parameter defines how far the influence of a single training example extends. A ‘low’ γ value corresponds to ‘far’ and a ‘large’ variance in the RBF kernel. I tested four values of C and γ , and three values of degree. Hence, for each vocab size I tested 84 models. I recorded the results from the best performing models for each vocab size.

The full parameter sets are shown in Fig 12 and the best performing model parameters for each vocab size along with accuracy scores are shown in Fig 13. The polynomial and RBF kernels performed best, the former with degree 3. The accuracy scores are close to each other, within five percentage points. The overall best performing model occurred with vocab size 100 and achieved accuracy of 82%. As well as accuracy I calculated other performance measures given by the following formulae using the counts of images classified as either True Positive (TP), False Negative (FN), False Positive (FP), or True Negative (TN) [9][10]. In this scenario TP corresponds to an image that is correctly classified as containing a zebra crossing, whereas FN corresponds to an image incorrectly classified as not containing a zebra crossing. The total number of images used in testing is equal to the sum of these counts (that is $TP+FN+FP+TN$).

$$\text{accuracy} = \frac{TP+TN}{TP+FN+FP+TN} \quad (3)$$

$$\text{recall (sensitivity)} = \frac{TP}{TP+FN} \quad (4)$$

$$\text{specificity} = \frac{TN}{TN+FP} \quad (5)$$

$$\text{precision} = \frac{TP}{TP+FP} \quad (6)$$

$$\text{MCC} = \frac{TP \times TN - FP \times FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}} \quad (7)$$

Fig 14 and 15 show the table and plot of each of these scores for the best performing models for each vocab size. The accuracy scores are all very good, around 80%, this is the simple percentage of correctly identified images out of all images used in testing. Precision, the proportion of correctly classified positive images out of all those classified as positive, is very high, over 85% for all models, and 95% for the over all best model with vocab size 100. This implies the

vocab_size	accuracy	recall	specificity	precision	mcc
5	0.780612	0.622449	0.938776	0.910448	0.591603
10	0.780612	0.612245	0.948980	0.923077	0.596033
20	0.770408	0.632653	0.908163	0.873239	0.562590
50	0.790816	0.704082	0.877551	0.851852	0.590586
100	0.821429	0.673469	0.969388	0.956522	0.672999
250	0.806122	0.704082	0.908163	0.884615	0.625407

Fig. 14. Table of scores for the best performing models for each vocab size.

model rarely classes a non-zebra crossing image as positive. This is confirmed by the equally high specificity scores; the proportion of correctly classified non-zebra crossing images out of all negative images.

The recall (sensitivity), the proportion of positive images correctly identified as containing zebra crossings, however, is less impressive, ranging from 61% to a maximum of 70%. On investigation of the images I realised my labelling method was not strict enough, and I had labelled images with only the barest slither of overlap with the zebra crossing polygons as positive samples. As noted above, with more time, I could easily rectify this to only label positive images if they meet a given threshold of overlap with the zebra crossing polygons. The relatively poor recall score reflects only my error in labelling the training and test sets. Since I used clean images, manually verified, for the *Letter Set* to build the vocabularies the basis for the image feature histograms is unaffected. In fact, the model has, helpfully, recognised my error; I have incorrectly told it that some images have zebra crossings, when in fact, they do not. I can only speculate that there must be relatively few of these researcher-introduced false positives. Logically there is a maximum of two per image, which would occur if the two metre sliding window happened to align with the edge of a zebra crossing. Fig 8 above shows an example of a zebra crossing spread over five images, with each image containing sufficient section of zebra crossing to warrant a positive label. The next image to the west (left) would not overlap at all with the zebra crossing, however the next image to the east (right) may include a thick slither of zebra crossing, and still be recognisable as such.

The MCC (Matthews Correlation Coefficient) score, is another measure of the quality of a binary classification, identically defined to Pearson’s phi coefficient [11]. Since it includes information from the recall score it has also been

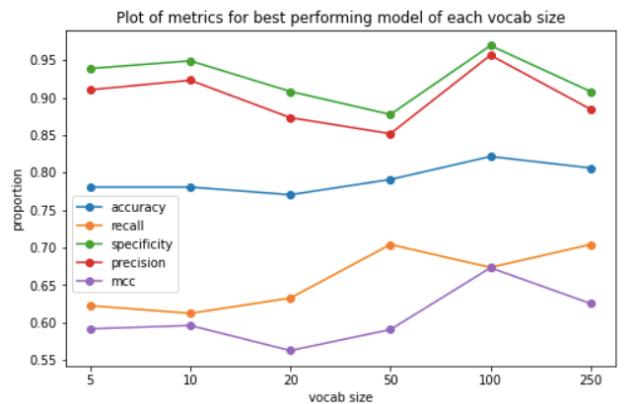


Fig. 15. Plot of scores for the best performing models for each vocab size.

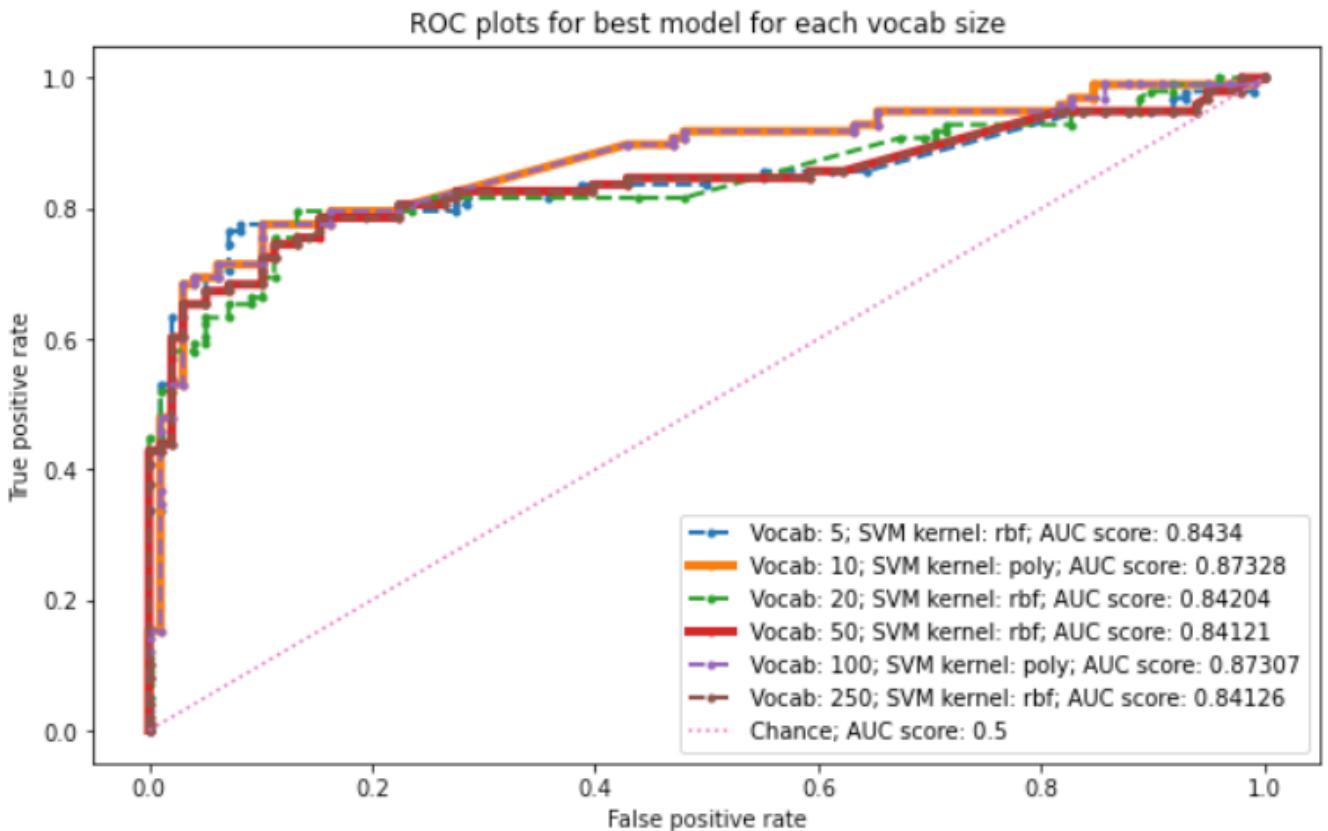


Fig. 16. ROC plot of best performing models for each vocab size, and corresponding AUC scores . The plots for vocab size 10 and 50 closely match those with vocab size 100 and 250, respectively. These are delineated with solid lines to show up under the dashed lines.

affected by my labelling error, and scores range from 56% to 67%. The best performing model on all metrics, apart from recall, is therefore the model using vocab size 100. However, it may be possible to achieve better results with lower vocab sizes given the generally high scores in specificity, precision and accuracy. Further research, with more accurately labelled images, is required.

Fig 16 shows a plot of the ROC (Receiver Operating Characteristic) curves for each model. This is generated using the probability per sample that it belongs in the positive class. The sample true positive rates (sensitivity/recall) are plotted against the false positive rate ($1 - \text{specificity}$), hence the closer the area under the curve (AUC) is to 1, the better the model [12]. The best AUC scores, of 0.87, are achieved by the models with vocab size 10 and 100, both of which use the polynomial kernel with degree 3. Considering the error in labelling these are very good results and confirm the viability of BoVW with SVM model for identifying zebra crossings.

Fig 17 shows some examples of image classifications that are TP, FN, FP and TN from the test results of the best performing model with vocab size 100. The first image in the TPs is an example of a badly labelled ground truth image, the zebra crossing polygon only intersects in the extreme top left corner, hence this should have been labelled as a negative sample. The rest are correctly labelled in the ground truth as well as correctly predicted.

The FN images include four poorly labelled ground truth examples; all except the fourth image should have been labelled as non-zebra crossings. This shows that the classifier is actually working correctly, despite the poor ground truth

labelling; it has predicted these as non-zebra crossings. As demonstrated in Fig 7 above, the clean images used in the *Letter Set* only contain the main stripes of zebra crossings and none of the surrounding dashes or zig-zagged lines.

The three FP images are interesting as it appears that white cars present a problem. Presumably they sufficiently resemble the features of the thick white stripes of the zebra crossing. The middle image is likely incorrectly classified due to the effects of the false positives in the ground truth. If there were enough present in the training set, and the vocab size large to allow for enough clusters for these to separate into a new group, then I speculate that this could introduce a rogue feature histogram interpreted as part of the positive class. I expect this histogram to be relatively flat since it would contain very few of the visual words from the vocab. However, if such a flat histogram was deemed to represent a positive image, this may cause problems. Further investigation of the feature histograms distributions might highlight such anomalies.

The TN images shows examples of good discrimination between the thick white stripes of zebra crossings and other white road markings such as the arrows indicating traffic calming humps in image one, and other white line configurations such as cross-hatching in image four.

Run time is a consideration with any machine vision model. I used a relatively small number of images in training (458) and testing (196), for a total of 654, and each image patch is small, only 32 by 32 = 1024 pixels. The number of images used was essentially dictated by how many zebra crossings happened to occur in the 82 km² of aerial



Fig. 17. A) True Positives; B) False Negatives; C) False Postives; D) True Negatives. Examples of five randomly selected images from the best performing model (vocab size 100) test results (C only contains 3 images in total).

photography available to me. The SVM grid search of parameters took about 10 minutes per vocab size, about 1 hour in total, to run all $6 \times 84 = 504$ tests using a modestly powerful four year old machine with one 2.2 GHz dual core CPU and 8GB RAM. Hence, I am confident that this method would be scalable and could therefore be used to create a dataset of all zebra crossings in the UK. This would be useful in contexts other than the main motivation for this work; to inform EV chargepoint infrastructure planning. For example, detailed information of road features, such as zebra crossings, could theoretically be used in future to support autonomous vehicle navigation as a cross referencing aid to onboard image processing software.

V. CONCLUSION

The main objective is to inform EV chargepoint infrastructure planning by researching ways of estimating parking potential in residential areas. This work has successfully explored the value of various datasets, devised a simple methodology to iteratively subtract non-parking areas from the known total length, or perimeter of residential roads, and researched some of the more complex subtraction tasks such as identifying road junctions and zebra crossings.

The advantage of an iterative approach is that one can choose which of the subtractions to apply. Some are notably simpler than others, for example, it is easier to subtract a fixed length to account for a bus stop as the bus stop location dataset already exists. The zebra crossing location dataset does not exist, hence the more complex task of creating it which I have made some progress to resolve.

The main limitation of the compactness approach to identifying road junctions is that concave shapes, such as cross-shaped road junctions have low compactness scores, as well as longer road sections, that are not road junctions.

The main limitations of the BoVW and SVM approach to identifying zebra crossings are the potential size of the data when scaling to the UK. Similar scale, UK wide projects have been done, such as the Green Spaces project to determine the proportion of green space in residential areas [13], to inform health and well-being, hence this not insurmountable.

The main impact of this work for the DfT team is to provide a review of datasets and techniques which may be useful to estimate on-street parking potential. Together with my previous project to estimate off-street parking, these constitute a useful resource, including practical coding examples. This will enable the team to carry out further studies, extend to the UK and crucially undertake the work in-house, thus maintaining full transparency and control of methodology, as well as saving public money.

REFERENCES

- [1] OLEV, “Making the Connection The Plug-In Vehicle Infrastructure Strategy,” 2011.
- [2] Wikipedia contributors, “Isoperimetric inequality,” *Wikipedia, The Free Encyclopedia*, 2021. [Online]. Available: https://en.wikipedia.org/wiki/Isoperimetric_inequality. [Accessed: 04-Aug-2021].
- [3] J. Flynn and C. Giannetti, “Using Convolutional Neural Networks to Map Houses Suitable for Electric Vehicle Home Charging,” *AI*, vol. 2, no. 1, pp. 135–149, 2021.
- [4] Emu Analytics, “ON-STREET EV CHARGER REQUIREMENTS PRODUCT SHEET,” 2018.
- [5] S. Se, “Zebra-crossing detection for the partially sighted Zebra-crossing Detection for the Partially Sighted,” *Proc. IEEE Comput. Soc. Comput. Vis. Pattern Recognit.*, no. May, 2014.
- [6] D. Koester, B. Lunt, and R. Stiefelhagen, “Zebra Crossing Detection from Aerial Imagery Across Countries,” *Int. Conf. Comput. Help. People*, vol. 9759, pp. V–VI, 2016.
- [7] Ordnance Survey, “AddressBase Plus Technical Specification,” vol. 117, no. 803, pp. 366–366, 2018.
- [8] APGB, “APGB Aerial Photography User Guide,” *APGB Support Knowledge Base*, 2020. [Online]. Available: <https://support.apgb.co.uk/en/article/apgb-aerial-photography-user-guide>. [Accessed: 25-Jul-2020].
- [9] Wikipedia contributors, “Precision and recall,” *Wikipedia, The Free Encyclopedia*, 2021. [Online]. Available: https://en.wikipedia.org/wiki/Precision_and_recall. [Accessed: 05-Oct-2021].
- [10] Wikipedia contributors, “Sensitivity and specificity,” *Wikipedia, The Free Encyclopedia*, 2021. [Online]. Available: https://en.wikipedia.org/wiki/Sensitivity_and_specificity. [Accessed: 05-Oct-2021].
- [11] Wikipedia contributors, “Matthews correlation coefficient,” *Wikipedia, The Free Encyclopedia*, 2021. [Online]. Available: https://en.wikipedia.org/wiki/Matthews_correlation_coefficient. [Accessed: 04-Oct-2021].
- [12] Wikipedia contributors, “Receiver operating characteristic,” *Wikipedia, The Free Encyclopedia*, 2021. [Online]. Available: https://en.wikipedia.org/wiki/Receiver_operating_characteristic. [Accessed: 05-Oct-2021].
- [13] C. Bonham, “Green spaces in residential gardens,” *Data Science Campus*, 2019. [Online]. Available: <https://datasciencecampus.ons.gov.uk/projects/green-spaces-in-residential-gardens/>. [Accessed: 12-Jun-2020].