IOSUD - Universitatea Politehnica Timişoara
Şcoala Doctorală de Studii Inginereşti

Urban Landmark Detection Using Computer Vision

Doctoral thesis – Abstract

for obtaining the scientific title of doctor at
Polytechnic University of Timisoara
in the Phd field ELECTRONIC ENGINEERING, TELECOMMUNICATIONS AND INFORMATION TECHNOLOGIES

author ing. Ciprian-Constantin Orhei
scientific leader Prof.univ.dr.ing. Radu Vasiu

month 05 year 2022

1 Motivation

Landmarks are typically defined from two perspectives: one as an object or structure that is easy visible and to recognize, and the second as a building or place that has an important historical importance. Landmarks in an urban area serve as “spatial magnet” in which cultural, civic, or economical activities take place. In this sense they have become an important aspect in multiple domains related to tourism and culture [1][2].

Identifying and locating of an urban landmark is an activity that naturally blends several research domains like image signal processing (ISP), computer vision (CV), augmented reality (AR). This blending of multiple domains was the first trigger that caused me to choose this research topic for the thesis.

The human interaction between landmarks and the ecosystem of the cities has become an interesting topic for me because of two events: one is the award of my hometown of Timişoara the title European Capital of Culture for the year 2023 and my interactions with the project Spotlight Heritage Timişoara.

As a result of this thesis, I wish to offer an urban landmark detection solution, from street view perspective, that can be utilized in a mobile solution for an AR tourism application. This direction desires to exploit the continuous development of user applications aimed for Timişoara European Capital of Culture 2023. Results of the proposed landmark detection system are presented in Figure 1.1.

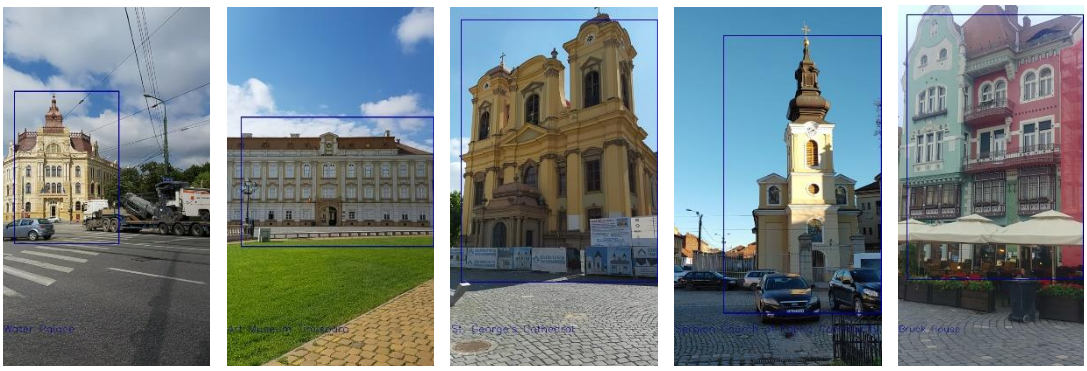


Figure 1.1 Example of proposed urban landmark detection

In this thesis I will attempt to answer the following research questions:

1. What is the state of the art in urban landmark detection using mobile cameras imaging?

2. What should a simulation framework offer to be considered as a suitable solution for processing systems of this nature?

3. What ISP algorithms enhance the image to obtain a better detection in this case?

4. What are the challenges in creating an urban landmark detection solution tailored for the Timişoara use-case?

The thesis is structured in several chapters that are described below.

Chapter 1 is an exposition of my motivation towards choosing the subject of this thesis. With the brief exposure I wish to explain the interconnections of multiple domains that founded the decision of choosing this research topic.

Chapter 2 offers an overview of the urban landmark detection domain, from general aspects focusing on the end to a specific sub-domain of content-based image retrieval system. The chapter focuses on presenting the domain ecosystem with all the challenges and solutions that literature has to offer.

Chapter 3 aims to present my chosen simulation system. The capability of offline simulating a system is an important one with considerable benefits in the development direction. End-to-End Computer Vision Framework (EECVF) [3][4] is an open source, python-based framework with the goal to offer a flexible and dynamic tool for researching.

Chapter 4 presents a proposed image sharpening algorithm that is low computational and based on dilated filters [5][6]. The proposed algorithm is evaluated on several use-cases that can appear in landmark detection system to better understand the benefits.

Chapter 5 presents the proposed landmark detection algorithm with a deep dive in each constructing block of it. I tried for each architectural decision inside the algorithm to explain and justify it in our given use-case context. The evaluation of the proposed landmark detection algorithm using popular dataset, presented in Chapter 2, plus the Timişoara specific dataset that was created for this scope.

Chapter 6 is the concluding part of the thesis. I start with some general conclusions regarding the research that I have done. Afterwards, I continue with enumerating theoretical and practical contributions that this thesis brings in the scientific fields.

2 Urban Landmark detection

Building or landmark recognition in urban environments aims to distinguish between different unique classes in a large-scale image dataset. This blend of technologies with the scope of landmark recognition is used in other several domains nowadays like computer gaming, urban planning, entertainment industry, movie making or digital mapping for orientation [7].

In general landmark recognition is a challenging task in the CV domain. Several challenges can occur when trying to fit several images from same place as changes in illumination conditions and viewpoint, or the presence of distractors such as trees, people, or signs. In order to mitigate these challenges, the existing approaches rely on feature description with a certain degree of invariance to scale, orientation and illumination [8].

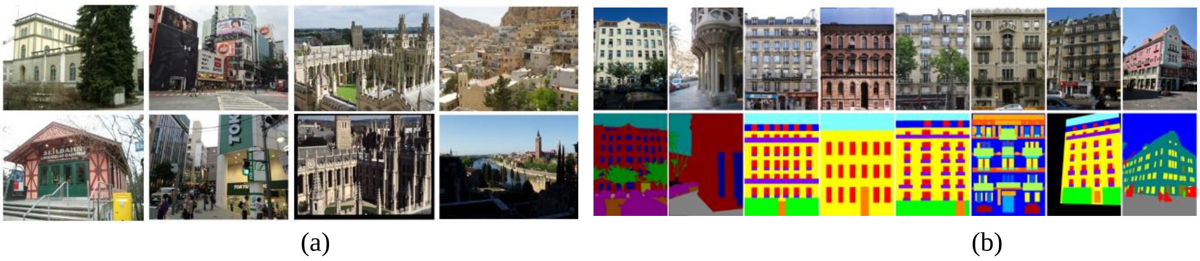


Figure 2.1 (a) Example of images from the building recognition datasets; (b) Example of images from the urban environment understanding dataset.

To better understand the challenges at hand I divided them in smaller problematics that I treated separately: datasets for landmark recognition, datasets for understanding the environment, detection solutions and specific challenges, benchmarking the detection.

Benchmarking a building detection system is not a trivial thing to do. This is a complex task due to differences and variety in both side, detection algorithm and benchmark scheme. Constructing an overall fitting benchmark is close to impossible due to different cues used in the detection pipeline or unicity of the landmarks used in the scope. Examples are presented in Figure 2.1(a).

The perspective of the images that create the dataset is important for many detection systems. Some detections schemes rely on the fact that the input images are from a street-view perspective. This assumption permits the designers of the system to employ specific calculations or pipeline decisions.

Automatic urban scene object recognition aims to classify and segment the image of an urban environment into categories like buildings, car, people, vegetation, and so on. Example of images from urban environment understanding datasets are presented in Figure 2.1(b). Nowadays, due to evolution of CV algorithm, typically scene understanding is done via semantic segmentation.

Ranking based metrics are present in literature and are growing more popular. In a sense, if we consider that the detection of classes (landmarks in our case) is not pure image-based problems the ranking metric is more appropriate. A metric that concerns highlighting the positive success of detection is the $TopK$, presented in Equation (2.1) [9].

$$\boldsymbol{TopK = \frac{1}{|Q|}\sum_{q\in Q} Relevant\ image\ retrivied\ in\ first\ K\ images} \qquad (2.\mathbf{1})$$

In Figure 2.2 I present the generic scheme for a landmark detection. As we can observe, the system will always include two phases: the offline phase, where we prepare the features vectors for training the classifier or classification model in case of convolutional neural networks (CNN) based solutions, and the online phase where we use one image to inquiry the classifier to obtain a detection. Some of the blocks detailed in Figure 2.2 are optional (marked with dashed lines) and they are decision that are made by the design of the system.

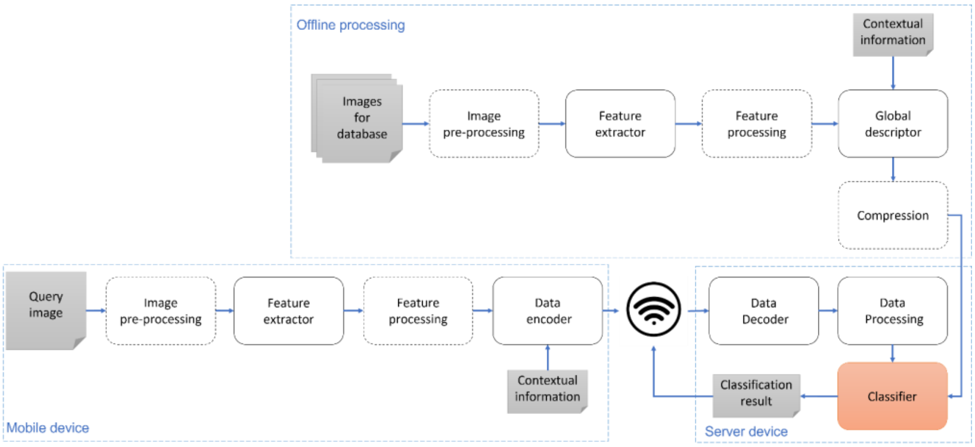


Figure 2.2 Generic scheme for landmark detection using mobile devices

In general, we see that solution proposed in literature are diverse from the feature point of view to the clustering and matching solutions. In the last years we can observe a clear CNN based trend in the solutions proposed. From the feature used point of view the range of solution proposed in literature vary from classical, so-called low-level features, as edge, junctions to global features, local features and CNN based solutions.

First real difficulty is the GPS location accuracy of the data that can drift in densely areas. The design of retrieval systems should handle and compensate for the error prone GPS tag system, even if geo-tagging has advance considerable in the last years.

Second challenge is the information that is captured by the images. In urban scenarios the quantity of distractors is considerable, and in this scope multiple solutions are already proposed for mobile landmark recognition systems.

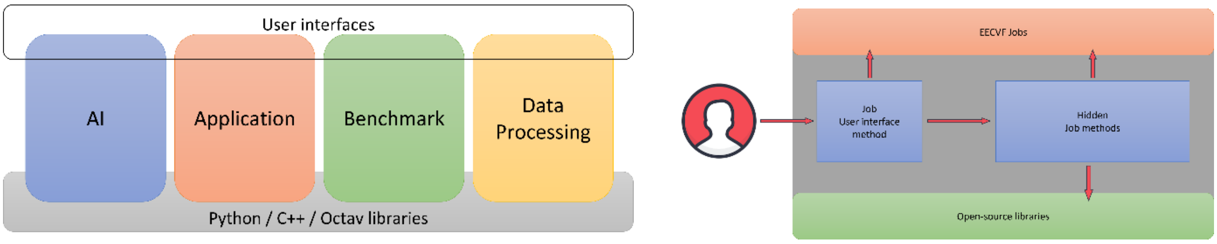
The third problem systems must tackle is the trade-off between recognition accuracy and real-time requirements constrains. Adding to the run-time and resource challenge is the bandwidth limitation of current mobile devices in telecommunication networks.

3 End-To-End Computer Vision Framework

CV pipelines include multiple steps starting from image acquisition from sensors, processing steps to enhance the image, transformation in order to reduce noise, selection of region of interest, segmentation of the image; different levels of feature extraction, high-level processing relevant to the application, and decision-making such as classifying an object [10].

To be able to simulate the proposed system I used the python-based CV framework EECVF [4]. The motivation of using this framework was based on several elements: the versatility of the programming language used, the capability of extending existing CV algorithms, the capability of debug information outputting. Naturally the familiarity of EECVF framework was an important factor but not singular.

Another interesting aspect I can point out is the main programming language used in the development of the frameworks. We can clearly see a corelation between the maturity level of Python language and the usage of it in constructing frameworks. This aspect is natural, from my opinion, as python offers build in function to interconnect several services provided by operating systems or HW accelerators.



(a) (b)

Figure 3.1 (a) EECVF construction blocks; (b) Job structure overview

The framework can handle the cycle of creating - training - evaluating an AI model, executing a CV application using the model and finally evaluating the results and displaying them without user intervention. From the capability of handling this chain the name of “End-to-End”.

EECVF is an easy to use, modular, and flexible framework designed for researching and testing CV concepts. The block overview is presented in Figure 3.1 (a). The framework does not require the user to handle the interconnections throughout the system [3]. This of course is not particular to this framework, but the trivial way in which new CV algorithms can be plugged in the EECVF is an important aspect in the decision.

Every job has a public interface, present in the User Interface block, from which it can be configured via parameters. This public method should handle all the necessary changes inside the blocks and even trigger other jobs, from same or different block, if necessary. In Figure 3.1(b) the internal structure of a job is presented.

One disadvantage of the EECVF is the lack of an GUI for basic users but in our use case this is not the case. Furthermore, my interest was more in developing and expanding new CV algorithms into the framework than just using existing ones.

4 Dilated filters

The first mention of dilation in image processing domain appears in the mathematical morphology field. The dilation operation uses a filter element to probe and expand the shapes contained in a image [11].

The second mention of the idea of dilated kernels appeared in the ML domain in recent years. It is a technique in which one expands the kernels by inserting holes between the consecutive elements [12].

The third mention of dilated filters in literature is the similar idea from ML but applied to classical filter-based convolutions outside of any AI model. The experiments were conducted on the edge detection kernels with good results [4], [5], [13].

To benefit from a higher neighbourhood of a pixels to obtain a pixel edge we define dilated filter as expanding the original filter by a dilation factor. When the kernels are dilated, the newly added positions are considered as gaps, and we ignore them by setting zeros [5]. The main objective of dilating is to cover more information from the input in the output obtained with every convolution operation. By applying this operation, it results in a wider field of view at the same computational cost.

Positive results were obtained by using dilated filters in several edge detection algorithms, work that was done in the past in [4] [5]. In Figure 4.4 visual results are presented when using one of the most popular classical edge detection algorithms: Canny [14]. As we can observe in the images, each level of dilation can bring with it extra edge points.

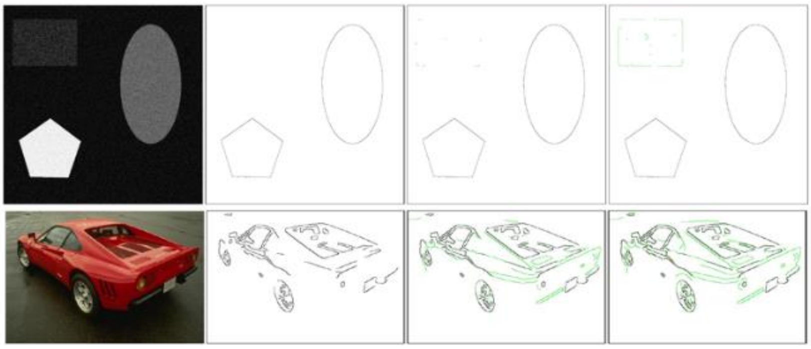


Figure 4.1 Edge map resulted using Canny algorithm with same parameters. Columns are original image, edge map using 3x3 kernel, using 5x5 dilated kernel, using 7x7 dilated kernel. Source of image is [9].

Encouraged by the positive results dilated filters bring to the edge detection process now I would like to investigate how this affects the sharpening process.

Image enhancement of contrast is a concern related to the sharpening of certain features like edges, object boundaries or textures. The main scope of this activity is to improve the visual appearance of the image.

Various approaches have been advanced for contrast enhancement. One of the most common methods used is the histogram equalization. Another popular solution for sharpening is the high-pass filter (HP). Filters like the low pass, band reject or HP are identified as ideal filters. An ideal filter has the property that it cuts all frequencies above (or below) a certain threshold frequency [15].

Another popular solution for image enhancement is the unsharp masking (UM) technique, this technique appeared in photography with the aim to improve the quality of pictures by masking their details. In this variant we would consider subtracting a blurred copy of the image from the original image itself. Even if the solution is a simple one it comes with several downside like highly sensitive to noise and possible enhancements to high-contrast areas [16].

In this scope we would dilate the following Laplace kernels found in literature [15]. By doing this we would like to take in consideration when apply the sharpening of a wider neighbourhood. To correct assess if this is a benefit, we need to take in consideration the 5x5 extended kernels.

$$\underset{\text{V1 }3\times3}{\begin{bmatrix}0&1&0\\1&-4&1\\0&1&0\end{bmatrix}}\quad \underset{\text{V1 }5\times5}{\begin{bmatrix}0&0&1&0&0\\0&1&2&1&0\\1&2&-17&2&1\\0&1&2&1&0\\0&0&1&0&0\end{bmatrix}}\quad \underset{\text{V1 }5\times5\text{ dilated}}{\begin{bmatrix}0&0&1&0&0\\0&0&0&0&0\\1&0&-4&0&1\\0&0&0&0&0\\0&0&1&0&0\end{bmatrix}}\quad \underset{\text{V2 }3\times3}{\begin{bmatrix}1&1&1\\1&-8&1\\1&1&1\end{bmatrix}}\quad \underset{\text{V2 }5\times5}{\begin{bmatrix}1&1&1&1&1\\1&1&1&1&1\\1&1&-18&1&1\\1&1&1&1&1\\1&1&1&1&1\end{bmatrix}}\quad \underset{\text{V2 }5\times5\text{ dilated}}{\begin{bmatrix}1&0&1&0&1\\0&0&0&0&0\\1&0&-8&0&1\\0&0&0&0&0\\1&0&1&0&1\end{bmatrix}}$$

Figure 4.2 Laplace kernels: $\boldsymbol{3x3}$, $\boldsymbol{5x5}$ and $\boldsymbol{5x5}$ dilated

The evaluation of an enhanced image is a difficult task for two reasons: there is no specific index that determines the optimal level of sharpness and the fact that any sharpening algorithm we would use has a several parameters that can change the output. From literature we concluded that the following metrics are used to evaluate and judging the effect of sharpening: entropy of an image, the spatial factor and mean contrast [17] [18].

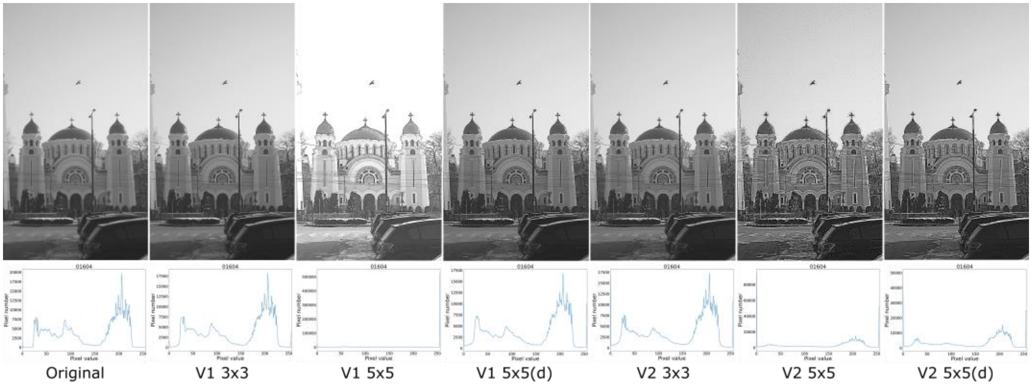


Figure 4.3 Visual results of HP using Laplace V1 and V2

In Figure 4.3 we are presented with a visual example of the results we obtained on image “01604” from the TMBuD dataset [19]. As we can observe when using the dilated filters, the images are enhanced better than with the classical 3x3 but lower than when using 5x5. From the histograms we can conclude that using V1 kernels does not flatten that much the histogram as in case of the V2 kernels.

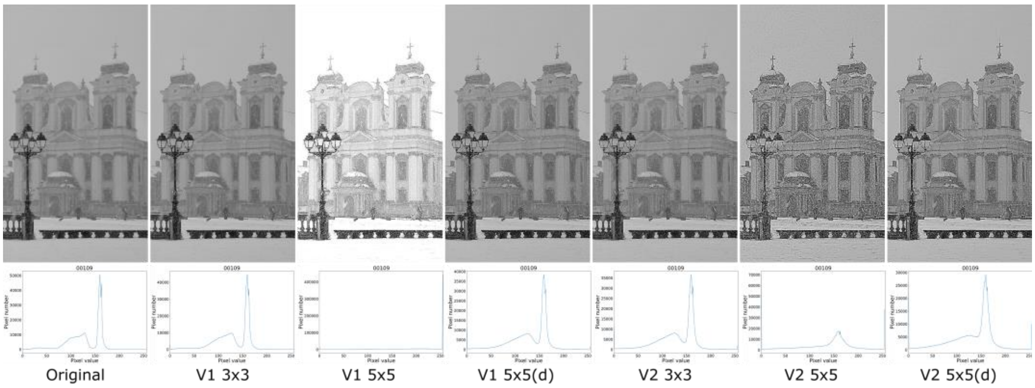


Figure 4.4 Visual results of UM using Laplace V1 and V2

In Figure 4.4 visual results are presented for image “00109” from the TMBuD dataset [19]. As before we can observe that using 5x5 dilate kernels bring forward better results than the classical 3x3. Another aspect worth mentioning is that using 5x5 extended filter does not seem to be a good solution for this activity.

Table 4.1 Average results per entire dataset for HP and UM

	HP		**UM**	
	Entropy	**SF**	**Entropy**	**SF**
Original image	7.105	14.881	7.105	14.881
HP V1 $3x3$	7.234	31.383	7.234	31.383
HP V1 $5x5$	5.143	69.286	5.143	69.286
HP V1 $5x5$ dilated	7.236	43.032	7.236	43.032
HP V2 $3x3$	7.226	49.896	7.226	49.896
HP V2 $5x5$	6.396	92.332	6.396	92.332
HP V2 $5x5$ dilated	6.966	62.801	6.966	62.801

From Table 4.1 I concluded that using dilated filters for image sharpening is a path worth pursuing. In our scope we only analysed the feasibility of the concept using standard basic classical sharpening solutions in order to prove this. This scope limitation is driven by a practical factor too, factor being that in the proposed landmark detection algorithm a fast computational solution is needed.

5 Proposed landmark detection systems

Landmark (building) recognition is deemed to be an object detection or content-based image retrieval problem for a specific scope. Compared to general object recognition task, this specific one brings more challenges because most urban image contain both human-made objects and natural ones.

Images taken of the same building could demonstrate a wide range of variability – they may be taken from different viewpoints, under different lighting conditions, or suffer from partial occlusions from trees, moving vehicles, other buildings, or themselves. Therefore, an ideal building recognition technique should be sensitive enough to identify an individual building while robust to different geometric and photometric image transformations [7].

The proposed algorithm is capable of handling difficult situations or scenarios when multiple landmarks of interest are clustered into a small arial like city squares or tourist neighbourhoods. These situations are the most interesting ones from the applications perspective as tourism application usually have difficulties in this kind of situations.

5.1 TMBuD – detection dataset

TMBuD building detection dataset is an extension of the TMBuD dataset [19]. This dataset aims to support a quantitative evaluation of mobile visual search supported by GPS context, of a subset of landmarks in Timişoara area. The main novelty of this dataset is the clear targeted scope of containing landmarks only from Timişoara city, which is the main arial targeted by our landmark detection system.

The TMBuD building dataset contains 1097 images with the resolution of 768x1024 pixels taken using a mobile phone from a street view perspective. The images are organized into 125 district landmarks located in several tourist parts of Timişoara.

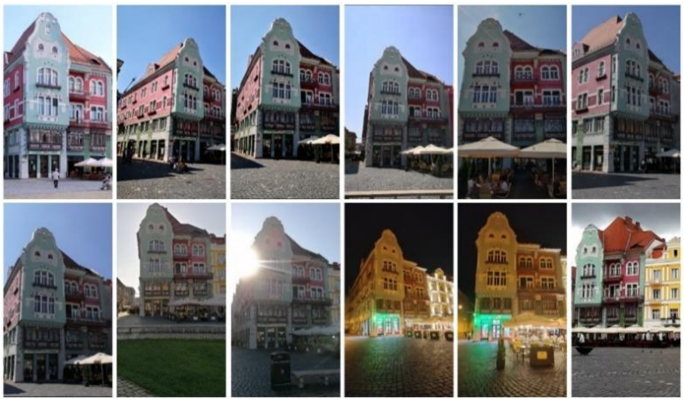


Figure 5.1 Example of one unique landmark inside TMBuD dataset

An important aspect to specify is that the benchmark comprises mobile photos of urban landmarks that aim to include variable quality, blurring, lighting changes, occlusions, and various viewing angles. This aspect is presented in Figure 5.1 where several images were selected from one landmark. The intention was to capture photos of different shot size (close, medium, long) and different angles of direction to be able to mimic real life scenarios.

The proposed benchmark dataset has different perspectives and conditions, as stated before, but it offers the possibility to evaluate a system in two unique conditions: low resolution images captured and night-time conditions. As presented in Figure 5.2 the dataset offers a limited number of images from landmarks in a low resolution, 7.66% of them, and images taken in the night, 14.9% of them.

To measure the photography diversity for each landmark we use the (lossless) JPEG compression size of the average images. This measure is done by computing the average image of each landmark and creating the lossless JPG file. The size of the file reveals the amount of information available. The theory behind this is that a diverse image will result in a blurrier average, compared with one that has a little diversity will result in a more structured, sharper one. Therefore, we can expect to see a smaller JPG file size of the average image for a landmark that offers more diverse images. The distribution across the dataset can be observed in Figure 5.2.

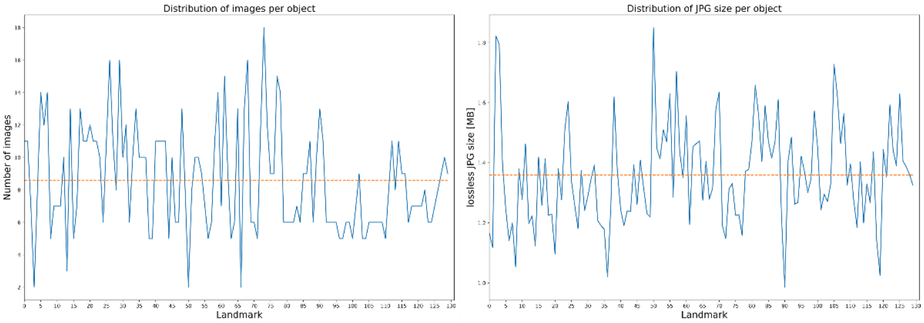


Figure 5.2 Number of images (left) and lossless JPG size (right) for each landmark from the dataset

5.2 Proposed detection system

In Figure 5.3 I present the overall processing pipeline proposed for my use case. The proposed flow is presented in detail in the future subchapter and evaluated afterwards.

If we want to summarize all the steps needed for this action, we can enumerate the following: (i) image and metadata gathering; (ii) image pre-processing (task needed to enhance the final classification); (iii) feature extractor; (iv) code book generation; (v) classifier.

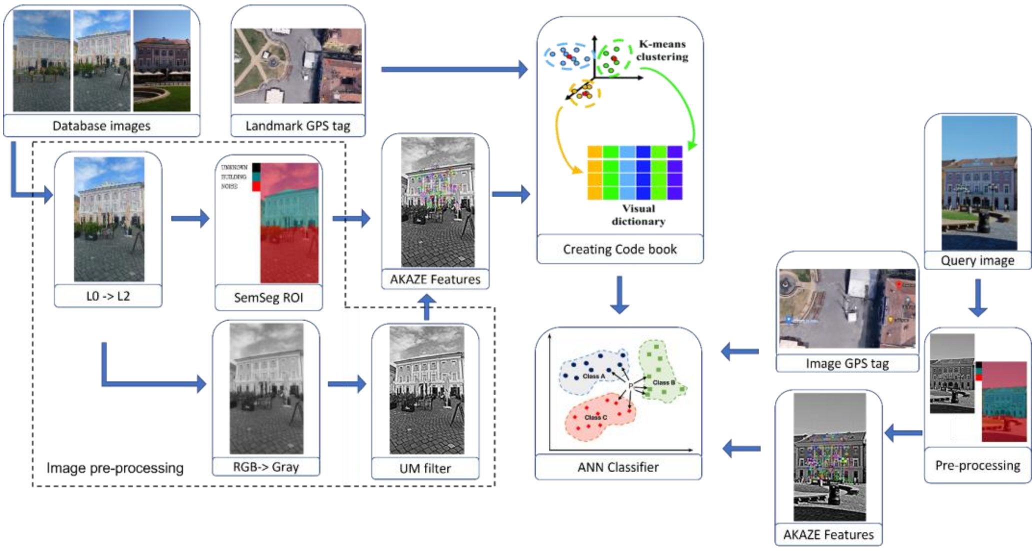


Figure 5.3 Proposed pipeline for landmark detection

The proposed system in the offline path takes the dataset images and generates a vector of A-KAZE features [20] that is filtered using a region of interest. This action compared to other solutions does not add a smaller weight to the features outside the region but ignores them all together. The assumption is that they are generated from distractors in the image. Afterwards, the features vectors are grouped together in a vector for each landmark which are clustered into BOF structure [21] using the KNN clustering algorithm [22].

In the on-line phase we mirror the pre-processing part in order to filter out the features that are generated from distractors and using ANN classifier [23] we find the closest similar landmark vector to the inquiry image.

To enhance the detection, we use GPS tags to limit our search range within the clusters. By doing so we gain benefits in the direction of detection accuracy and run-time.

In order to generate better feature, in term of localization and quantity, we enhance the images using dilated sharpening UM algorithm. The configuration of the UM is carefully chosen so the low-quality images get enhance but the good quality images do not get over-sharpen.

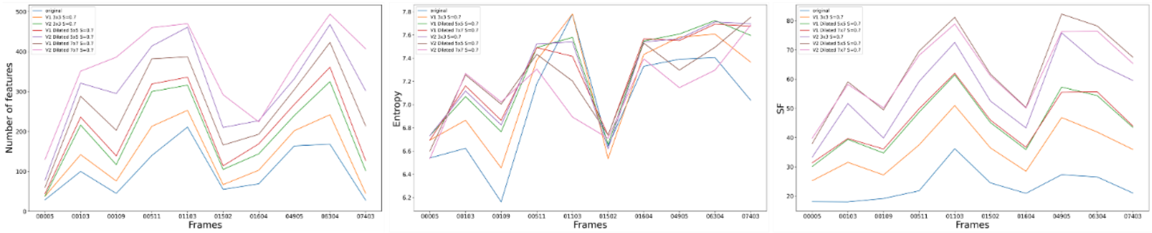


Figure 5.4 Number of features and image proprieties for pyramid level 2

In Figure 5.4 we are presented with the results for the entire dataset used. We can observe that in every case sharpening the image resulted in bigger number of features detected. More, when using dilated filters, the number increased even more. From the statical data presented, I can conclude that the benefits of sharpening the image using dilated filter before feature extractions step brings further benefits regardless the pyramid level that is processed.

For our system I investigated an DL based method for choosing our region of interest, so we will aim to achieve a classical semantic segmentation. For our system I chose Residual Networks with 50 layers (ResNet-50) as base model [24] of our network with SegNet [25] as an segmentation model

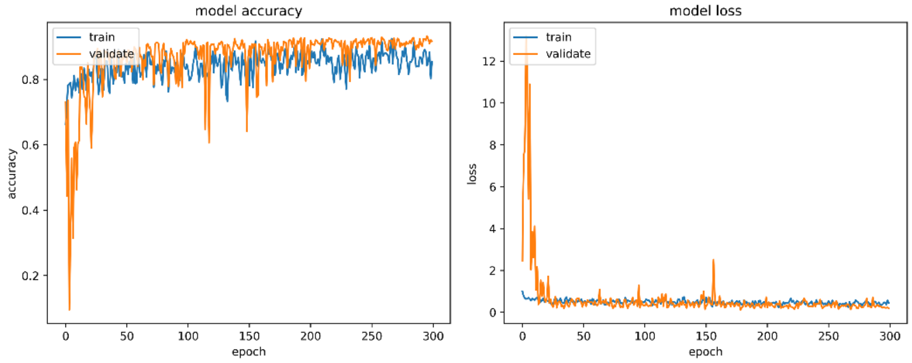


Figure 5.5 ResNet50-SegNet training model metrics

In Figure 5.5 I present the plotted values for Accuracy and Loss for the training process. The two graphs show the sanity and correctness of the training process. The first graph tracks the accuracy, and it offers a valuable insight on the amount of overfitting that is happening in the model. If the gap between the training and validation graph is high than the model is strongly overfitted. The second graph tracks the training loss, as evaluation on the individual batches is done during the forward pass. The loss can become more linear with a low learning rate. With high learning rates they will start to look more exponential.

5.3 Benchmark the proposed detection system

For this chapter, I will use the following notations: A-KAZE descriptor size (D_S), A-KAZE number of octaves (D_NO), A-KAZE number of layers (D_NL), A-KAZE threshold (D_THR), smoothing dilated filter size (UM_D), UM strength of smoothing (UM_S), BOF cluster size (B_S), threshold for distance for feature matching (THR), distance in m between GPS tags (D_GPS).

As metric for our evaluation, I have chosen to use the Top1 metric, presented in Equation (1). If we consider the landmark detection system, more than CBIR this metric makes more sense than mAP.

5.3.1 Benchmark on public dataset

First, I would like to evaluate the performances of the proposed algorithm on public datasets to have a comparison with another existing algorithm. From the analysis done in chapter 2.2, on datasets, and the summary done in Table 2.4 I choose to evaluate our proposed system on two popular datasets: ZuBuD [26] and ZuBuD+ [8].

For this experiment I scaled all the ZuBuD images to 240x320 size. This transformation of the input images was done by other solutions found in literature. The scope of this scaling is to ease the processing.

As we observe in Table 5.1 we obtain good results on ZuBuD and ZuBuD+ datasets with our proposed algorithm. For this evaluation of the proposed system the GPS tag processing is disabled because the dataset used do not have that information. But we experimented with different configuration of pyramid level and ROI selection.

Table 5.1 Top1 results on dataset

Solution	Config	ZuBuD	ZuBuD+
[8]	-	100.00%	100.00%
Proposed no ROI on L0	D_S=8, D_NO=5, D_NL=6, D_THR=0.0012, UM_D=7, UM_S=0.1, B_S=300, THR=0.82	99.13%	99.80%
[27]	-	99.00%	99.00%
[28]	-	99.00%	-
Proposed on L0	D_S=8, D_NO=5, D_NL=6, D_THR=0.001, UM_D=7, UM_S=0.5, B_S=150, THR=0.80	98.26%	99.60%
[29]	-	98.00%	-
[30]	-	95.00%	-
[31]	-	94.00%	-
Proposed on L1	D_S=8, D_NO=4, D_NL=6, D_THR=0.001, UM_D=7, UM_S=0.5, B_S=45, THR=0.80	93.91%	98.60%
Proposed no ROI on L1	D_S=8, D_NO=4, D_NL=6, D_THR=0.001, UM_D=5, UM_S=0.9, B_S=30, THR=0.82	92.17%	98.20%

5.3.2 Benchmark on proposed dataset

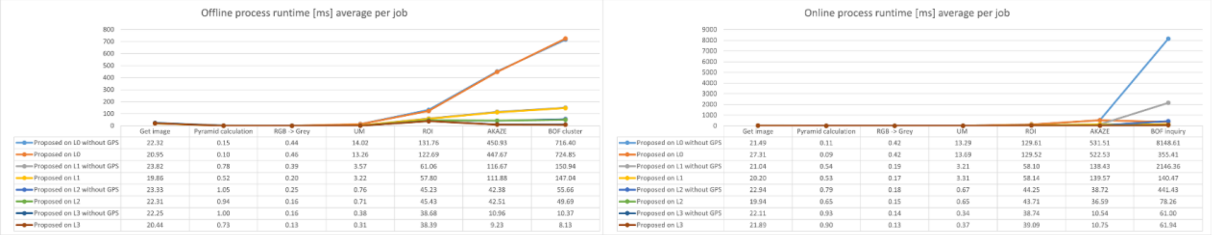
In this section I want to evaluate the proposed algorithm on the novel TMBuD dataset, which is presented in chapter 5.2. In order to have a better understanding I use all three configurations available: Dataset 3_2 (3 views for training and 2 for evaluating), Dataset 3_5_Night (3 views for training and 5 for evaluating that contain night scenarios images), Dataset 3_N (unbalanced number of test images).

Table 5.2 Summary of experiments done with Top1 results

Configuration	Lvl	GPS	3_2	3_5_Night	3_N
D_S=8, D_NO=6, D_NL=3, D_THR=0.001, UM_D=7, UM_S=0.9, B_S=400, THR=0.8, D_GPS=100	L1	Yes	93.62%	93.59%	92.05%
D_S=8, D_NO=6, D_NL=3, D_THR=0.001, UM_D=7, UM_S=0.9, B_S=400, THR=0.8, D_GPS=100	L0	Yes	91.91%	91.81%	91.91%
D_S=8, D_NO=6, D_NL=3, D_THR=0.001, UM_D=7, UM_S=0.9, B_S=350, THR=0.8, D_GPS=100	L2	Yes	92.82%	92.17%	90.10%
D_S=8, D_NO=6, D_NL=3, D_THR=0.001, UM_D=7, UM_S=0.9, B_S=400, THR=0.8	L1	No	88.44%	87.90%	85.65%
D_S=8, D_NO=6, D_NL=3, D_THR=0.001, UM_D=7, UM_S=0.9, B_S=400, THR=0.8	L0	No	82.86%	87.90%	84.66%
D_S=8, D_NO=6, D_NL=3, D_THR=0.001, UM_D=7, UM_S=0.9, B_S=350, THR=0.8	L2	No	86.85%	83.27%	79.78%
D_S=8, D_NO=6, D_NL=3, D_THR=0.001, UM_D=7, UM_S=0.9, B_S=30, THR=0.8, D_GPS=75	L3	Yes	74.90%	72.60%	69.59%
D_S=8, D_NO=6, D_NL=3, D_THR=0.001, UM_D=7, UM_S=0.9, B_S=30, THR=0.8	L3	No	48.61%	43.77%	41.42%

In Table 5.2 I present all the configurations and Top1 metrics obtained when evaluating the proposed dataset on different variants of it. As first preliminary conclusion I can observe that when introducing the GPS tag results are improved for any pyramid level. The second preliminary conclusion is the fact that using L3 pyramid level does not obtain good results overall even if the resource consumption is reduced dramatically. The lack of results on L3 is not something that we did not expect as the input image is only 90x160 pixels in size.

In Figure 5.33 I present with the run-time analysis for each configuration evaluated in Table 5.5. When talking about a detection system most of the time the decision comes done to a trade-off between resources and performance. First thing that we can observe is the fact that for bigger resolution images, like L0 and L1, the GPS tag filtering has a visible effect on overall online runtime. If we look on L2, which would be our target as we consider a good balance between performance and resources. Another aspect that we need to mention is that only by filtering using GPS we reduced the BOF inquiry runtime by 70%.



	Get image	Pyramid calculation	RGB -> Grey	UM	ROI	AKAZE	BOF cluster
Proposed on L0 without GPS	22.32	0.15	0.44	14.02	131.76	450.93	716.40
Proposed on L0	20.95	0.10	0.46	13.26	122.69	447.67	724.85
Proposed on L1 without GPS	23.82	0.78	0.39	3.57	61.06	116.67	150.94
Proposed on L1	19.86	0.52	0.20	3.22	57.80	111.88	147.04
Proposed on L2 without GPS	23.33	1.05	0.25	0.76	45.23	42.38	55.66
Proposed on L2	22.31	0.94	0.16	0.71	45.43	42.51	49.69
Proposed on L3 without GPS	22.25	1.00	0.16	0.38	38.68	10.96	10.37
Proposed on L3	20.44	0.73	0.13	0.31	38.39	9.23	8.13

	Get image	Pyramid calculation	RGB -> Grey	UM	ROI	AKAZE	BOF inquiry
Proposed on L0 without GPS	21.49	0.11	0.42	13.29	129.61	531.51	8148.61
Proposed on L0	27.31	0.09	0.42	13.69	129.52	522.53	355.41
Proposed on L1 without GPS	21.04	0.54	0.19	3.21	58.10	138.43	2146.36
Proposed on L1	20.20	0.53	0.17	3.31	58.14	139.57	140.47
Proposed on L2 without GPS	22.94	0.79	0.18	0.67	44.25	38.72	441.43
Proposed on L2	19.94	0.65	0.15	0.65	43.71	36.59	78.26
Proposed on L3 without GPS	22.11	0.93	0.14	0.34	38.74	10.54	61.00
Proposed on L3	21.89	0.90	0.13	0.37	39.09	10.75	61.94

Figure 5.6 Runtime analysis of proposed algorithm on different levels and configurations

If we look at the online processing runtime, we can observe that the system can run around 6 frames per second (total runtime is 160ms without image acquire job). We eliminated the job that obtains the image as in a real time system the process is different from the one, we used in our experiment. If we consider that in this step of the development the algorithm is not optimized for real-time conditions, it is a decent frame rate to process at.

From Table 5.1 and Table 5.2 we can clearly state that the proposed algorithm has good results obtaining a value of 99.13% Top1 on ZuBuD dataset and 92.05% on TMBuD v3_N dataset.

6 Contribution and conclusions

The background research for this thesis was done as part of the Multimedia Research Centre, Faculty of Electronics, Telecommunications and Information Technologies of the Politehnica University of Timişoara. The research activity was complemented by volunteer activities and contributed to mobile AR applications serving the city of Timişoara for the title of European Capital of Culture in 2023.

My thesis can be summarized as a proposal for a landmark detection scheme tailored for Timişoara's urban environment. This complex algorithm can be integrated in a mobile application that can offer tourists the chance to better discover the urban scenario of our city.

Next, I attempted to answer the research questions that I raised at the beginning of the thesis, based on my research findings:

1. **What is the state of the art in urban landmark detection using mobile cameras imaging?**

An extensive analysis and review of the landmark detection domain is provided in chapter 2. In order to be able to consider all the difficulties we can encounter, the problem was divided in smaller parts: landmark datasets and challenges, urban environment understanding datasets and landmark detection solutions.

2. **What should a simulation framework offer to be considered as suitable solution for processing systems of this nature?**

A review of existing framework solutions is provided in chapter 3 that focuses in highlighting main points of each framework and in which programming language they were developed. As criterion for choosing a framework I considered the following: (1) main programming language used; (2) Capability of simulating an entire chain of processing; (3) Capability of outputting intermediate and debug data; (4) Facile integration of new CV algorithms.

3. **What image signal processing algorithms enhance the image to obtain a better detection in this case?**

In order to answer this question, I turned to the concept of dilated filters, that are described in chapter 4. From literature I found that using dilated filters can improve the image processing pipeline. This concept was validated from filter-based edge detection to CNN semantic segmentation. In that sense I successfully applied the dilated concept to image sharpening algorithms and obtained better results with same processing resources needed.

4. **What are the challenges in creating an urban landmark detection solution tailored for Timişoara use-case?**

This is the fundamental question that was answered in this thesis, and it is tackled in chapter 5. As stated, before this is a two-part problem: the specific dataset needed for this and the tailored landmark recognition system. Starting with the information gathered and structured in chapter 2 I was able to provide a Timişoara dataset of landmarks that is easily scalable in the future and a landmark recognition system which can be facile adapted for a mobile application.

In the thesis I have the following theoretical contributions:

1. Overview of existing CV software frameworks
2. Analysis of dilated filters in several CV topics
3. Analysis on edge detection algorithm focused on urban scenarios
4. Overview of Building (Landmark) recognition datasets
5. Overview of Urban environment understanding datasets
6. Overview of Landmark recognition solutions
7. Theoretical bases for Timişoara urban detection

In the thesis I have the following theoretical contributions:

1. EECVF framework development
2. Classical Edge detection algorithm
3. Dilated High Pass and Unsharp Masking
4. Implementation in EECVF of the experiments done in the thesis
5. TMBuD recognition dataset
6. Semantic segmentation training and evaluation for Timişoara environment
7. Landmark recognition system for Timişoara urban environment

BIBLIOGRAPHY

[1] S. Vert, D. Andone, A. Ternauciuc, V. Mihaescu, O. Rotaru, M. Mocofan, C. Orhei, and R. Vasiu, “User Evaluation of a Multi-Platform Digital Storytelling Concept for Cultural Heritage,”

Mathematics, vol. 9, no. 21, Art. no. 21, Jan. 2021, doi: 10.3390/math9212678, WOS:000750692000001 (Q1 journal)

[2] K. Lynch, The Image of the City. MIT Press, 1964

[3] C. Orhei, M. Mocofan, S. Vert, and R. Vasiu, “End-to-End Computer Vision Framework,” in 2020 International Symposium on Electronics and Telecommunications (ISETC), Nov. 2020, pp. 1–4. doi: 10.1109/ISETC50328.2020.9301078, WOS:000612681000017.

[4] C. Orhei, S. Vert, M. Mocofan, and R. Vasiu, “End-To-End Computer Vision Framework: An Open-Source Platform for Research and Education,” Sensors, vol. 21, no. 11, Art. no. 11, Jan. 2021, doi: 10.3390/s21113691, WOS:000660684600001 (Q1 journal).

[5] V. Bogdan, C. Bonchis, and C. Orhei, Custom Dilated Edge Detection Filters. Václav Skala - UNION Agency, 2020. doi: 10.24132/CSRN.2020.3001.19.

[6] C. Orhei, V. Bogdan, C. Bonchis, and R. Vasiu, “Dilated Filters for Edge-Detection Algorithms,” Appl. Sci., vol. 11, no. 22, 2021, doi: 10.3390/app112210716, WOS:000725536600001 (Q2 journal).

[7] J. Li, W. Huang, L. Shao, and N. Allinson, “Building recognition in urban environments: A survey of state-of-the-art and future challenges,” *Inf. Sci.*, vol. 277, pp. 406–420, Sep. 2014, doi: 10.1016/j.ins.2014.02.112.

[8] F. Magliani, N. M. Bidgoli, and A. Prati, “A location-aware embedding technique for accurate landmark recognition,” in Proceedings of the 11th International Conference on Distributed Smart Cameras, New York, NY, USA, Sep. 2017, pp. 9–14. doi: 10.1145/3131885.3131905.

[9] H. Müller, W. Müller, D. McG. Squire, S. Marchand-Maillet, and T. Pun, “Performance evaluation in content-based image retrieval: overview and proposals,” Pattern Recognit. Lett., vol. 22, no. 5, pp. 593–601, Apr. 2001, doi: 10.1016/S0167-8655(00)00118-5.

[10] R. Klette, Concise Computer Vision. London: Springer London, 2014. doi: 10.1007/978-1-4471-6320-6.

[11] J. S. J. Lee, R. M. Haralick, and L. G. Shapiro, “Morphologic Edge Detection,” *IFAC Proc. Vol.*, vol. 19, no. 9, pp. 7–14, Jun. 1986, doi: 10.1016/S1474-6670(17)57504-7.

[12] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, “DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 4, pp. 834–848, Apr. 2018, doi: 10.1109/TPAMI.2017.2699184.

[13] C. Orhei, V. Bogdan, and C. Bonchiş, “Edge map response of dilated and reconstructed classical filters,” in *2020 22nd International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC)*, Sep. 2020, pp. 187–194. doi: 10.1109/SYNASC51798.2020.00039, WOS:000674702000028.

[14] J. Canny, “A Computational Approach to Edge Detection,” IEEE Trans. Pattern Anal. Mach. Intell., Nov. 1986, doi: 10.1109/TPAMI.1986.4767851.

[15] R. Jain, R. Kasturi, and B. G. Schunck, Machine vision. New York: McGraw-Hill, 1995.

[16] G. Ramponi and A. Polesel, “A Rational Unsharp Masking Technique,” J. Electron. Imaging, vol. 7, pp. 333–338, 1998.

[17] K. De and V. Masilamani, “Image Sharpness Measure for Blurred Images in Frequency Domain,” Procedia Eng., vol. 64, pp. 149–158, 2013, doi: 10.1016/j.proeng.2013.09.086.

[18] J. P. D. Villiers, “A comparison of image sharpness metrics and real-time sharpening methods with GPU implementations,” 2010.

[19] C. Orhei, S. Vert, M. Mocofan, and R. Vasiu, “TMBuD: A Dataset for Urban Scene Building Detection,” in *Information and Software Technologies*, Cham, 2021, pp. 251–262. doi: 10.1007/978-3-030-88304-1_20.

[20] P. F. Alcantarilla, A. Bartoli, and A. J. Davison, “KAZE Features,” in Computer Vision – ECCV 2012, Berlin, Heidelberg, 2012, pp. 214–227. doi: 10.1007/978-3-642-33783-3_16.

[21] T. Joachims, “Text categorization with Support Vector Machines: Learning with many relevant features,” in Machine Learning: ECML-98, Berlin, Heidelberg, 1998, pp. 137–142. doi: 10.1007/BFb0026683.

[22] D. Nister and H. Stewenius, “Scalable Recognition with a Vocabulary Tree,” in 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR’06), Jun. 2006, vol. 2, pp. 2161–2168. doi: 10.1109/CVPR.2006.264.

[23] J. H. Friedman, J. L. Bentley, and R. A. Finkel, “An Algorithm for Finding Best Matches in

Logarithmic Expected Time," ACM Trans. Math. Softw., vol. 3, no. 3, pp. 209–226, Sep. 1977, doi: 10.1145/355744.355745.

[24] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," 2016, pp. 770–778. Accessed: Jan. 18, 2022. [Online]. Available: https://openaccess.thecvf.com/content_cvpr_2016/html/He_Deep_Residual_Learning_CVPR_2016_paper.html

[25] V. Badrinarayanan, A. Kendall, and R. Cipolla, "SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation," IEEE Trans. Pattern Anal. Mach. Intell., vol. 39, no. 12, pp. 2481–2495, Dec. 2017, doi: 10.1109/TPAMI.2016.2644615.

[26] H. Shao, T. Svoboda, and L. Van Gool, "Zubud-Zurich buildings database for image based recognition." Technique Report No. 260, Swiss Federal Institute of Technolog, 2003.

[27] H. Jégou, M. Douze, C. Schmid, and P. Pérez, "Aggregating local descriptors into a compact image representation," in 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Jun. 2010, pp. 3304–3311. doi: 10.1109/CVPR.2010.5540039.

[28] C. Orhei, L. Radu, M. Mocofan, S. Vert, and R. Vasiu, "CBIR for urban building using A-KAZE features," in 2021 IEEE 27th International Symposium for Design and Technology in Electronic Packaging (SIITME), Oct. 2021, pp. 218–221. doi: 10.1109/SIITME53254.2021.9663587.

[29] D. M. Chen, S. S. Tsai, V. Ch, G. Takacs, J. Singh, and B. Girod, "Tree histogram coding for mobile image matching," 2009.

[30] W. Zhang and J. Košecká, "Hierarchical building recognition," Image Vis. Comput., vol. 25, no. 5, pp. 704–716, May 2007, doi: 10.1016/j.imavis.2006.05.016.

[31] N. J. C. Groeneweg, B. de Groot, A. H. R. Halma, B. R. Quiroga, M. Tromp, and F. C. A. Groen, "A Fast Offline Building Recognition Application on a Mobile Telephone," in Advanced Concepts for Intelligent Vision Systems, Berlin, Heidelberg, 2006, pp. 1122–1132. doi: 10.1007/11864349_102.