# EXTRACTING CURBSIDE STORM DRAIN LOCATIONS FROM STREET-LEVEL IMAGES

BY

ELIZABETH E. DEPWE

THESIS

Submitted in partial fulfillment of the requirements
for the degree of Master of Science in Civil Engineering
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2015

Urbana, Illinois

Adviser:

Professor Joshua Peschel

## ABSTRACT

This thesis presents a machine vision procedure to identify and extract storm drain locations from natural images along surface street curbsides. Existing storm drain infrastructure information is commonly reposed by managing agencies in either paper or digital format. Access to these data for urban hydrologic and hydraulic modeling purposes may be limited by security protocols and/or the format in which the data may be available. The procedure described in this work uses a novel vision algorithm with Google Street View imagery to identify and extract the locations of curbside storm drains. Results are converted into a tabular format that can be converted into geometric input files for modeling purposes. This fast, approximation approach to assembling storm drain data could be of interest to public works managers, urban hydrology and hydraulics practitioners and researchers, and citizen scientists, to improve general understanding of the civil and environmental infrastructure.

## ACKNOWLEDGMENTS

First, I would like to express my thanks to my adviser, Professor Joshua M. Peschel for providing the opportunity for me to pursue this research. Without his support this work would not have been possible.

I would like to thank Dr. Derek Hoiem for his time and advice. His insight on computer vision feature detection was tremendously beneficial to the development of this work.

I am grateful to my friends and colleagues in Urbana-Champaign who have been an invaluable asset to me throughout my thesis. A special thanks to Meng Han, Adam Burns, Tianyu He, Christopher Chini, Zachary Barker, Matthew Czapiga and many more. Their encouragement and assistance has significantly contributed to my efforts in this project and my experience as a whole at the University of Illinois.

Lastly, I would like to express my deepest gratitude to my family who have consistently shown me patience, kindness and gentleness as I work on this project. Their love never fails to remind me to seek out the good things.

# Table of Contents

# CHAPTER 1: INTRODUCTION

Stormwater runoff management in the urban environment is primarily accomplished using an underground conveyance network. These systems are designed to quickly collect excess water and deliver it to a secondary location where it can be treated or directly returned to the ecosystem [1]. Without a functioning runoff management system roadways close, and homes and businesses can flood. These systems become even more critical to community health and safety when stormwater infrastructure is used to transport wastewater (e.g. combined sewer overflow systems). As of 2004, combined wastewater and stormwater systems can be found in over 32 states and are responsible for an estimated 850 billion gallons of combined sewage discharge per year [2]. Storm drains are the entry points to these systems. The strategic location and continual maintenance of inlets are crucial to the ability of the network to collect and manage runoff. Furthermore, improvement to and understanding of infrastructure is often accomplished first through modeling of current infrastructure. How and where the network accepts water is integral to the success of this modeling process.

Despite the essential role these inlets play, their location and condition are many times unknown. Documentation of storm drains within a conveyance network is usually conducted by a municipality; however, the resolution of the data can vary greatly, excluding key inlets in critical areas. Inexact location data may impact, the ability to understand how the system as a whole is functioning. An example of a discrepancy in realistic and mapped drain locations is shown in Figure 1.1. This figure shows storm drain location data from the City of Champaign in Illinois and the actual location of storm drains which were manually collected. Some of the data points coincide but 75% of drains were not found in the City of Champaign data set. The purpose of this study is to create a tool which maps the location of storm water inlets in a computationally efficient manner. The tool first allows the user to specify an area of interest and downloads street-level images for this area from Google Street View (GSV). Each downloaded image is then processed using a computer vision algorithm. A csv file containing latitude and longitude is exported from the processing algorithm corresponding to the location of the positively identified drains. The following section investigates previous studies which collect

street-level images for analysis and studies which use computer vision to identify data relevant to the field of civil engineering.


**Figure 1.1: Available data from the City of Champaign mapping the location of storm drains compared to the actual location of storm drains. In the region shown, only 25% of the storm drains are represented in the city data set.**

## 1.1 Why Google Street View (GSV)?

Since 2007, GSV has been a publically available source of 360-degree panoramic images along roadways in over 20 countries around the world [3]. The images are saved based on their latitude and longitude and are accessible through an HTTP request [4]. The extent of the dataset and the ease of access makes GSV a practical source for infrastructure data.

Street-level imagery has found an application in many different research fields. Rundle et al. studied the accuracy of using freely available street-level images to assess neighborhood characteristics such as aesthetics and pedestrian safety [5]. The audits evaluated detailed metrics and found that 54.3% of the characteristics assessed with GSV highly correlated with typical field audit results. It was also reported that the street-level imagery approach saved 33 hours per auditor. While the results from GSV auditing were not identical to those from typical field assessment, the time saved is non-trivial.

GSV panoramic street-level imagery was successfully utilized by Xiao et al. to create 3D models of facades and street-scapes [6] [7]. These studies found that the imagery provided by GSV was accurate and detailed enough to create realistic 3D representations of the infrastructure. Identification and assessment of infrastructure was completed by Balali et al. using GSV imagery to map and categorize the type of traffic signs within each image [8]. Similarly, Hara et al. detected the presence of curb ramps using images sourced only from GSV [9].

This collection of studies address a variety of needs within the engineering community for understanding the state of infrastructure and cataloguing specific data related to this infrastructure all using publicly available data from GSV

## 1.2 Computer Vision Applications in Civil Engineering

Computer vision techniques vary in their logical approach, computational expense and accuracy. Machine learning techniques are based on the training of an algorithm given a set of images that are correctly characterized by the user. Partial training was used by Hara et al. to detect curb ramps within a set of Google Street View images [9]. Machine learning, crowdsourcing, and feature extraction techniques were all applied to create a reliable detection tool. This study suggested that the best results of machine learning could be obtained from limiting training images to those found within the exact region sampled. Although machine learning has the potential to be extremely accurate, it's success is dependent on very robust algorithmic coding and the correct selection of training images.

Rai et al. also used a computer vision technique including feature extraction, machine learning and user response to determine human preferences related to greenness within a neighborhood. Feature extraction filtered the images based on their color histogram, shape boundaries, spatial histogram, and openness of a scene. It was found that machine learning played a key role in adapting the algorithm to pick out attributes dictated by the user.

Edge detection is a computer vision technique which detects localized changes within an image [10]. Different algorithms for identifying edges were analyzed by Abdel-Qader et al. in an study to identify cracks in concrete bridges [11]. The work tested the fast Haar transform, fast Fourier transform, Sobel and Canny algorithms. For the application of detecting cracks, the fast

Haar transform was found the most reliable: however, it was noted that each edge detector varied in its strengths.

The development of a computer vision algorithm to identify and classify traffic signs within GSV images was conducted by Balali et al. [8]. The method used included feature extraction through histogram orientation gradients and machine learning. Conclusions from this study show that shape and color paired with the trained machine learning algorithm were successful at detecting traffic sign infrastructure.

In addition to the identification of specific features, extensive work has been conducted by Hoiem et al. on the establishment of general geometric context within an image [12] [13] [14]. These studies characterize an image into three planar features, segmenting ground objects from vertical objects and sky. Uniquely this geometric classification is done without the assistance of machine learning and is based upon the data found in only one image.

## 1.3 Contributions

The main contributions of this work are the potential to map storm drain location instantly using publically available data, the use of GSV as a data source, and the application of color and feature based computer vision methods for identifying storm drain features. More accurate data for storm drain locations would improve the design of new infrastructure and the improvement of existing infrastructure. Assessing the usefulness of GSV images as a data source is a key component to this contribution. While independent street-level video data could be accomplished, utilizing the GSV database expands the application and shortens the timeline of data collection. Finally, the application of only color and feature based computer vision methods identifies the extent to which constant processes can be applied to a set of images without input from the user or machine learning techniques.

## 1.4 Outline of the Thesis

The chapters of this work are organized as follows:

- Chapter 2 consists of a review of the methodology used. It first details the approach to acquiring street-level images then details the approach used to process these images.
- Chapter 3 explains how the methodology was implemented. Examples of the tools and algorithms utilized to accomplish the methodology are detailed.
- Chapter 4 summarizes the results of applying the methodology to a test case. It details the images acquired and the processing algorithms applied.
- Chapter 5 discusses the implication of the result as well as formative observations.
- Chapter 6 states the conclusions of this work. Future improvements to the project are also noted in this section.

# CHAPTER 2: METHODOLOGY

This section provides an overview of the methodology to extract storm drain location from street level imagery. The approach consisting of two major steps, retrieval of the desired images and processing of the images for storm drain identification and localization. (Figure 2) This methodology results in a comma-separated (csv) text file that contains the latitude and longitude approximated within the images where a drain was positively detected and identified. This file is formatted to be easily imported into a geographic information system (GIS) software for visualization and further analysis. The following sections describe the individual steps illustrated in Figure 2.1.

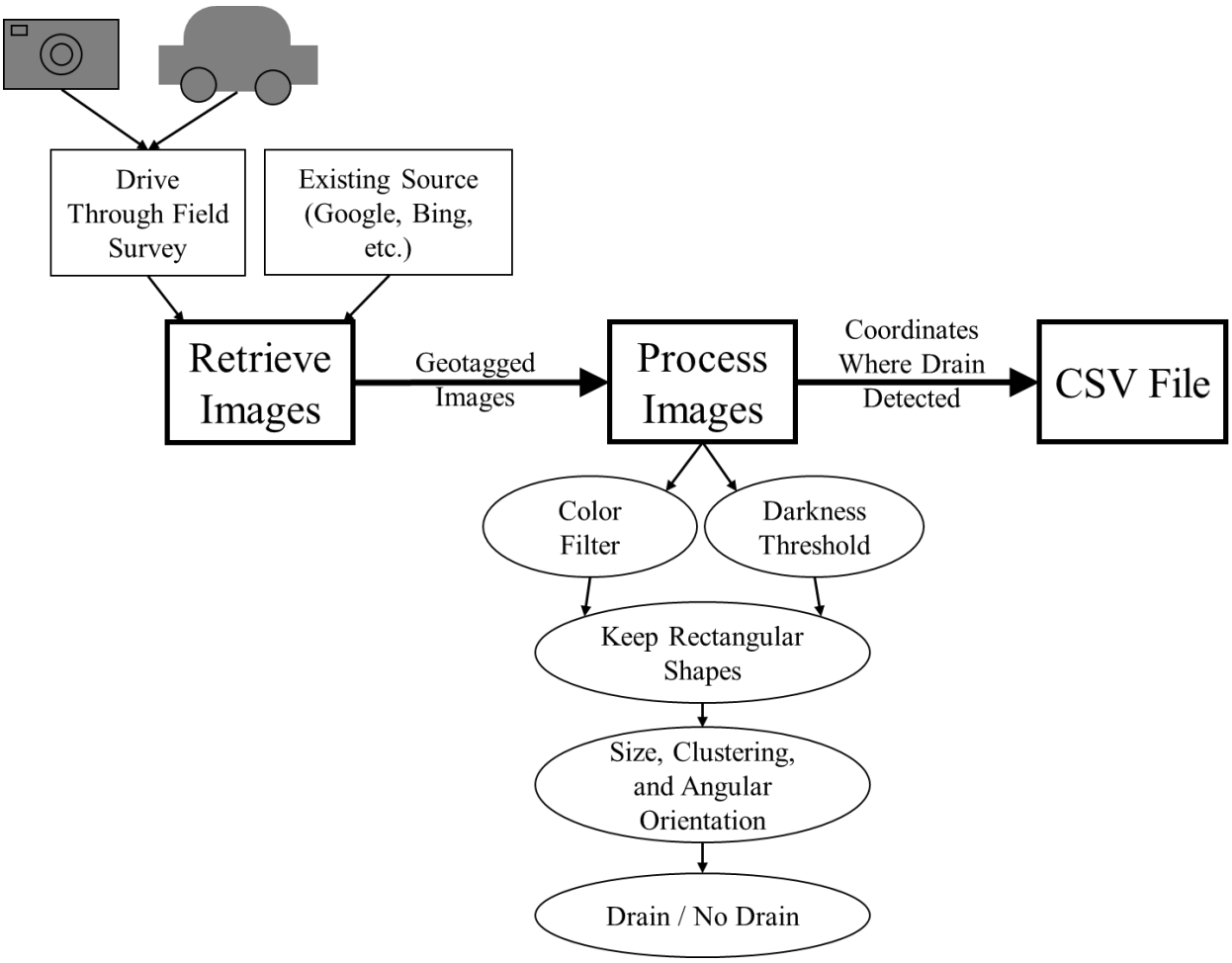



**Figure 2.1: A map of the project methodology for retrieving street-level imagery, processing the images to identify storm drains and exporting coordinates of identified drains.**

## 2.1 Acquisition of Street Level Imagery

Street level imagery can be obtained through a variety of sources and processes. It is reasonable to assume that any person or machine (e.g., a car) could independently collect images for a region of interest using a video or still image camera. Customized collection of image data would allow the entity to control factors such as image size and perspective. Another source of street level images is freely available through web services such as Google Street View (GSV) and Bing. These sources provide a large repository of instantly available image data. However, there can be limitations such as restrictions on the number of images downloaded in a given session, spatial resolution (number of pixels), and acquisition factors such as time of day the

images are taken (e.g., inconsistent shadows). While these limitations could be improved through custom collection of street level images, the factors do not present significant barriers for the identification of storm drains. In this work, the street level imagery was chosen to be retrieved from the existing database, GSV, due to the well-populated database and the ease of programmatic request.

The quality of the original imagery can play an important role in the ability of the processing algorithm to correctly detect the presence of a storm drain within the image. GSV images have a maximum resolution of 640x640 pixels for a majority of the locations covered; however, some remote areas may be further limited in size. If a "Google Maps API for Work" account is purchased, images are available up to 2048x2048 pixels [4]. All of the images tested in this study were at a resolution 640x640 pixels to be consistent with free public access. Image perspective can also be an important factor in determining the usefulness of the data for an object of interest within the image obtained from GSV. Perspective can be described by three parameters pitch, field of view and heading; the following sections describe and provide recommendations for these parameters.

### 2.1.1 Pitch

Pitch indicates the angle down or up relative to parallel with the roadway. If the pitch is set at +45° the image is angled toward the sky, if -45° the image is angled toward the roadway, and if 0° the image is parallel with the roadway. Figure 2.2 shows a schematic of the pitch parameter and Figure 2.3 shows a set of example GSV images with a pitch of +45°, -45° and 0°.

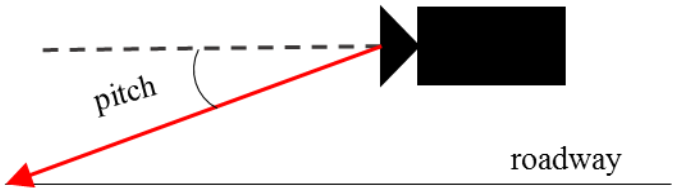



**Figure 2.2: Elevation view of schematic of how GSV defines the pitch parameter.**


**Figure 2.3: GSV images demonstrating the change in image perspective as a result in change of the pitch parameter. Images have a pitch of (a) +45°, (b) -45° and (c) 0°.**

### 2.1.2 Field of View

The field of view parameter describes the zoom or horizontal zone of interest. Field of view ranges from 0° to 120°, where 0° is a narrow perspective representing a small area and 120° is the widest perspective available. Figure 2.4 is a schematic of the field of view parameter and Figure 2.5 shows an example of GSV images with a field of view of 0° and 50°, respectively.

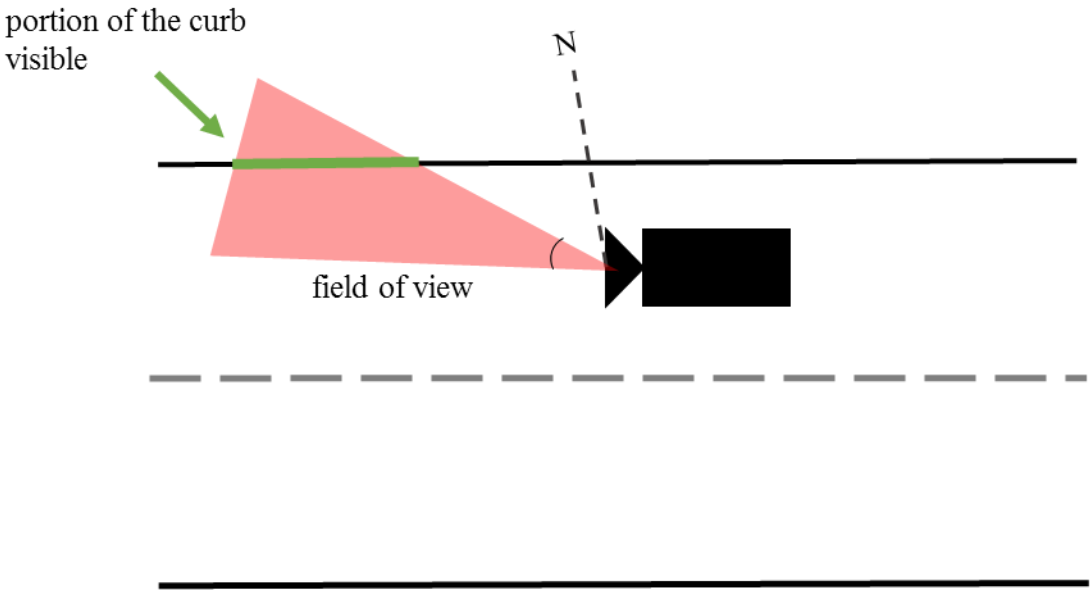



**Figure 2.4: Plan view schematic of how GSV defines the field of view parameter.**

(a) (b)

**Figure 2.5: GSV images demonstrating the change in image perspective as a result of change in field of view parameter. Images have a field of view of (a) 0° and (b) 50°.**

### 2.1.3 Heading

The heading parameter is a value 0° to 360° and represents the compass heading relative to north. Heading is independent of the roadway direction. Figure 2.6 shows a schematic of the heading parameter. To retrieve images with a heading consistent with the directionality of the roadway, the direction of the road relative to North must first be determined. Two GSV images with the same heading of 0° are shown in Figure 2.7. Although both images have the same heading, one shows a perspective perpendicular to the lane of travel and the other parallel. This is because the roadway in image (a) is oriented in an East-West direction, while the road in image (b) is oriented North-South direction. A perpendicular, downstream and upstream heading was downloaded for each side of the roadway. This ensured that all visible portions of the curb could be processed for each coordinate point along the road. Figure 2.8 demonstrates the six heading perspectives downloaded from GSV.

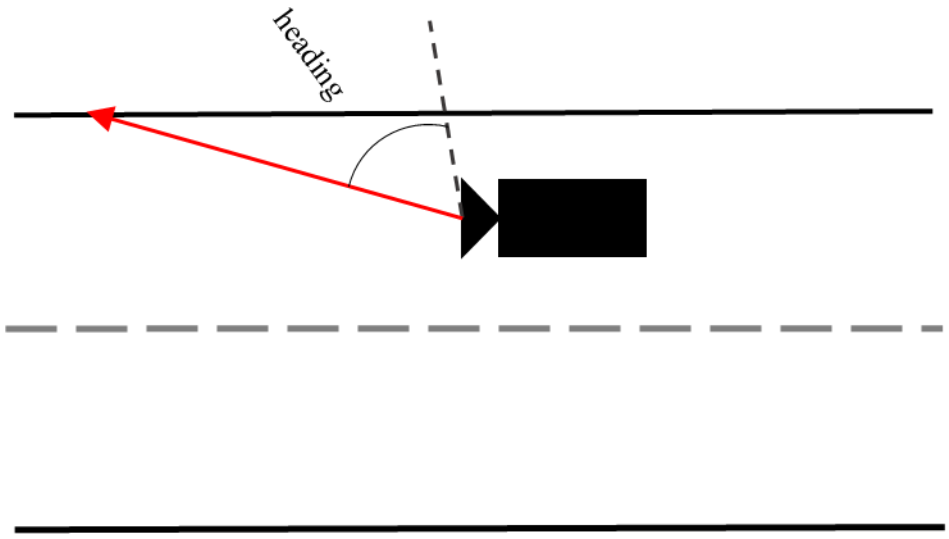



**Figure 2.6: Plan view schematic of how GSV defines the heading parameter.**

(a) (b)

**Figure 2.7: GSV images demonstrating that the heading parameter is relative to North and therefore independent of the directionality of the roadway. Both images (a) and (b) have a heading of 0°.**

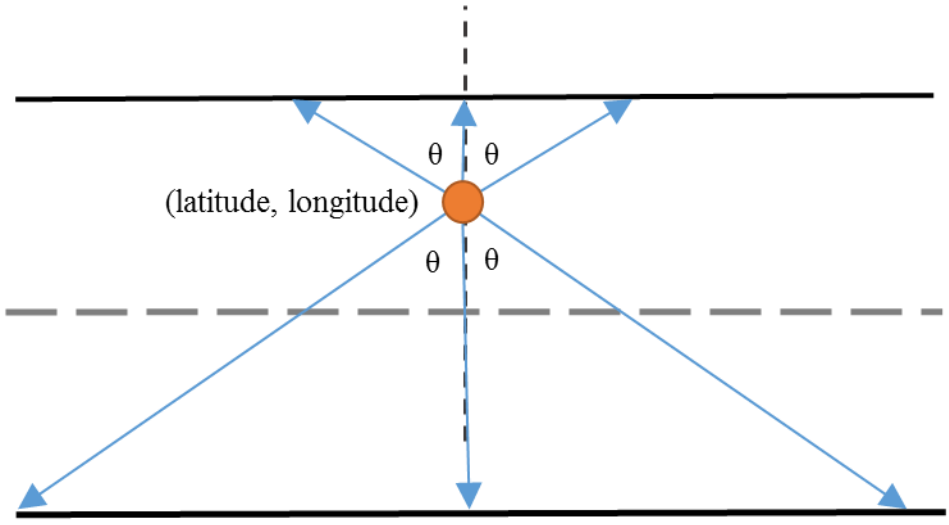



**Figure 2.8: Plan view schematic of the six heading perspectives downloaded for each GSV coordinate point.**

### 2.1.4 Retrieval Limitations

Aside from the size limitations of each image previously mentioned, GSV places limits on the number of images downloaded. A free API key was created to download a significant number of images. Limits on downloading without a key are not documented but became obvious during implementation. Requesting more than 25,000 images daily for 90 consecutive days with an API key may lead to interruption or cancelling of future requests. However, Google claims to be lenient on these limits [4].

## 2.2 Processing Images

Processing is the step that determines if an image contains storm drains. The original GSV images may contain extensive object and color variability due to environmental factors as well as perspective. Images retrieved from an intersection contain different geometric features than those from the middle of a block. Furthermore, images taken in a suburban environment can vary greatly from those taken in a dense urban street. Taking this variability into account, four filters were selected to evaluate the images: greenness, lightness, rectangularity, and geometric.

### 2.2.1 Filters

The first filter applied, greenness, is based on the color values comprising each pixel. Colors found in human made infrastructure are difficult to distinguish. Storm drain color values are not unique and a variety of similar values can be found in sidewalks, brick streets, trash cans, siding on buildings, etc. However, storm drains along curbsides are typically devoid of green making them have a distinguishable color and a practical application for a color based filter.

Storm drains have significantly darker features within an image along curbsides. The lightness filter sorts out the dark and light regions of the image saving only the dark regions for further analysis. After greenness and lightness is filtered the remaining regions are converted to polygons and filtered for their rectangularity. Storm drains have sharp manmade edges and should closely subscribe to a rectangular shape. Once the regions have been converted to rectangles, each rectangle is run through a series of geometric filters that test their geometric attributes as well as their spatial relation to a second rectangle and its geometric attributes. This final step removes very small and very large rectangles (AREA), removes rectangles that are not near to another rectangle (NEARNESS), removes rectangle pairs that are not parallel (PARALLEL), and removes rectangle pairs that are wider than they are tall (H_W_RATIO). Section 4.2 describes how these filters were applied and the threshold values chosen.

### 2.2.2 Additional Filters

In an effort to eliminate non-road data within the image, a processing filter to identify the curb was attempted. This proved to be a non-trivial task. Edge-detection algorithms such as "canny" and "hough" were highly affected by the shadows present within the images. Histogram equalization was attempted to neutralize the effect of the presence of shadows but proved instead to give equal significance to all edged features in the image. This increased the reliability of curb identification by edge-detection in some images but reduced it in other, namely those with heavy shadowing. Road detection has been studied extensively for other fields of research such as autonomous vehicle navigation [15] and the identification of planar features such as road, building face and sky [13]. Assuming that removal of non-road features is possible, some images were manually processed prior to the application of the aforementioned processing

filters. An example of this editing is shown in Figure 2.9. Results are reported for raw GSV images and manually pre-processed images Chapter 4 of this paper.

(a) (b)

**Figure 2.9: (a) Original GSV image and (b) manually edited image with non-road features removed.**

### 2.2.3 Saving Data

The output of the processing step is a binary result, either drain or no drain present. Only the coordinates of images with positive drain detection are saved. Since six images are analyzed for each set of coordinates, the saved, positive detection coordinates were adjusted to correctly communicate the side of the street on which the drain is located and if it is up or down the street from the location at which the image was taken. Through a trial-and-error process, an adjustment factor of ±0.000017° is applied for the latitude coordinate and ±0.000021° is applied for the longitude coordinate. Figure 2.10 shows the adjusted coordinates compared to the original coordinates used to download the images through GSV.

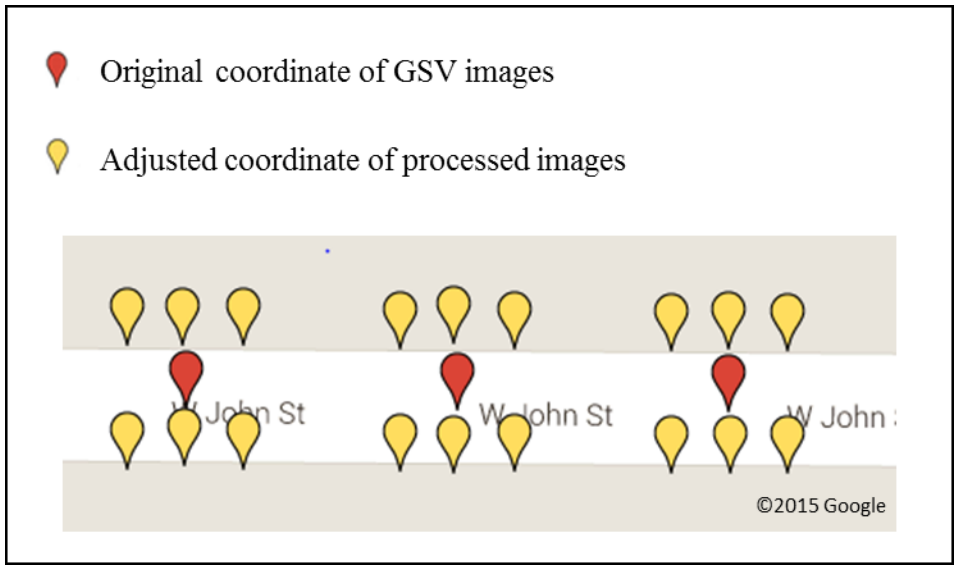



**Figure 2.10: Schematic of image coordinates adjusted from GSV source coordinate point based on a change in perspective.**

# CHAPTER 3: IMPLEMENTATION

A case study was conducted on West John Street in Champaign Illinois between Neil Street and Prospect Avenue. This region was chosen because it represents a generic residentially zoned area and because the data could be verified easily through field inspection. It should be noted that the region contains small shady roadways, larger open roadways, and occasional bike lanes and street parking. The region primarily consists of residential homes set back from the street with front lawns.

## 3.1 Retrieving Implementation

Image retrieval from GSV was completed through a GUI written in Java version 1.7 using the development environment Eclipse version 4.2. Java was chosen for this portion of the study because of its default libraries related to creating the user interface. The GUI allows a user to input the top right and bottom left corners of a bounding box and download a seamless stream of street level images. It also provides user input for image perspective parameters pitch, field of view and heading. Figure 3.1 shows a screenshot of the GUI.

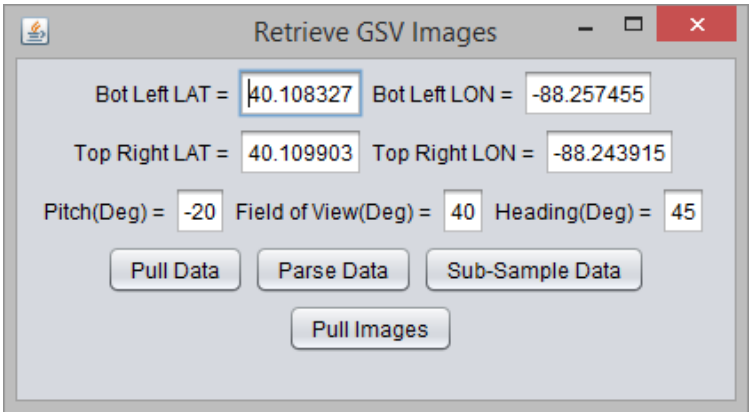



**Figure 3.1: Screenshot of Java GUI used to retrieve images from GSV.**

Before requesting images through GSV the coordinates of the roadways within the bounding box must be determined. Using the input bounding box, an http request is sent to OpenStreetMap which returns an xml file linking features such as roadways, bicycle paths,

buildings, parking lots, and lot delineation with a list of latitude and longitude coordinates. OpenStreetMap is an open-source mapping database and is subject to editing and the addition of variable features. A sample of the http request is http://overpass.osm.rambler.ru/cgi/xapi_meta?map?bbox=longitude1,latitude1,longitue2,latitude2 [16].

The data pulled from OpenStreetMap is then parsed so that only the coordinates linked to features with the tag "highway" are saved. This tag was found to correspond to all paved roadways on which cars travel. It does not include ally roadways which run between residential blocks or paved bicycle paths detached from vehicle roadways. The resulting parsed coordinates correspond to intersections and significant change in roadway curvature, and are tied to independent roadway ids making it possible to identify a set of coordinate points belonging to each road. The final manipulation of the OpenStreetMap data is sub-sampling. In order to consistently collect images along a roadway, coordinate points were added at an interval of no greater than 0.0001 decimal degrees. This value was found to correspond closely to the unknown interval at which GSV stores its 360-panoramic images. If the interval is decreased, images are duplicated, if the interval is increased, blind spots are present along the roadside. An example of the coordinate points before and after sub-sampling are shown in Figure 3.2. After sub-sampling, the list of coordinate points are ready to be used to download images from the GSV database.

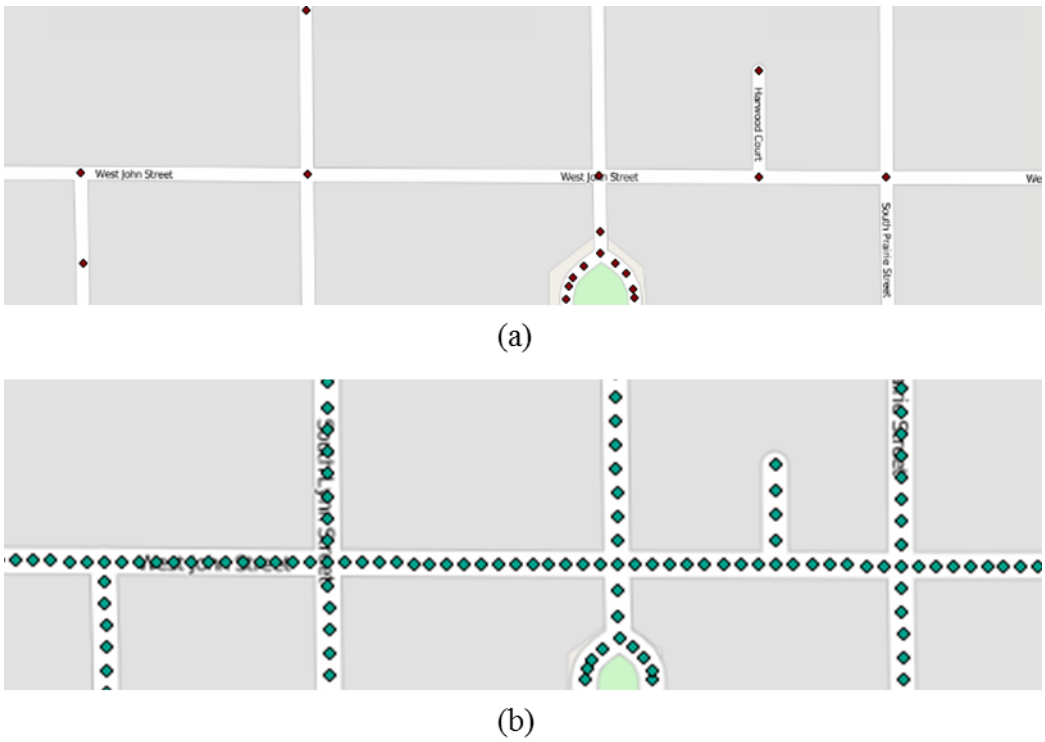



**Figure 3.2: Example of subsampling step when retrieving images. (a) Original coordinates from OpenStreetMap and (b) coordinates after sub-sampling.**

Pulling images is accomplished through an http request with several parameters which dictate the perspective of the image at each coordinate point. The parameters, as mentioned in Section 2.1, are pitch, field of view and heading. Images are requested at a size of 640x640 pixels which is the largest available for users without a paid Google Maps for Work account. A sample of the http request is https://maps.googleapis.com/maps/api/streetview?size=640x640&location=40.108327,-88.257455&fov=100&heading=90&pitch=-20.

For this study, pitch was set at a constant value of -25°. This value was determined empirically through visual assessment of the images downloaded. If the pitch is too severely directed toward the ground, the depth in the image is lost. Additionally, there is a dead zone in which the image detail is blurred and useless directly under the perspective of the camera, making a more severe pitch unprofitable.

In order to reduce unnecessary details in each image while avoiding blind spots along the curb, all the images downloaded were given a constant field of view of 50°. This value was

determined through inspection of downloaded images. Field of view and pitch are closely related. Ideal pitch values will influence the ideal field of view value and vice versa.

Finally, an ideal heading was determined to be an angle of 45° relative to the direction of the roadway. In order to download an image with this constant perspective, the angle of the roadway relative to north was required. Thus the angle between each set of coordinate points along a roadway was first determined and a heading of 45° was computed using this value.

## 3.2 Processing Implementation

Processing of the images is accomplished through C++ and executed using the Qt development interface. An open source computer vision library OpenCV is utilized extensively for image manipulation. C++ was chosen because of its efficiency invoking member functions and virtual functions [17] which are integral to the execution of this algorithm. It was also chosen because of its compatibility with the existing OpenCV libraries. The Qt interface was chosen because it was able to utilize the compiled OpenCV library. This attribute proved to be non-trivial when choosing an environment for development. As mentioned in the methodology, each image was processed by four filers; greenness, lightness, rectangularity and geometric.

Before greenness is removed, the whole image is converted from the red, green, blue (RGB) color space to the hue, saturation, value (HSV) color space. This is done using the function "cvtColor" within OpenCV. This command redefines the individual values contained in each pixel using the algorithm shown in Figure 3.3.

$$V \leftarrow \max(R,G,B)$$

$$S \leftarrow \begin{cases} \dfrac{V-\min(R,G,B)}{V} & \text{if } V \neq 0 \\ 0 & \text{otherwise} \end{cases}$$

$$H \leftarrow \begin{cases} 60(G-B)/(V-\min(R,G,B)) & \text{if } V=R \\ 120+60(B-R)/(V-\min(R,G,B)) & \text{if } V=G \\ 240+60(R-G)/(V-\min(R,G,B)) & \text{if } V=B \end{cases}$$

$$\text{If } H < 0 \text{ then } H \leftarrow H+360$$

**Figure 3.3: Algorithm from OpenCV for conversion between RGB and HSV color space [18].**

Once the image is converted to HSV, a threshold is applied removing all pixels which have a hue value less than 38 or greater than 75. These values were chosen based on empirical observation. After the threshold is applied, the remaining pixels are smoothed. The empty regions are then converted to polygons by filling convex contours and finally subtracted from the original image. Figure 3.4 shows an original GSV image and the image after the hue threshold is applied. The white areas within the processed image indicate areas containing green which will be removed from the original image.

(a) (b)

**Figure 3.4: (a) Original GSV image and (b) image after filter to identify greenness is applied.**

Similar to the process of removing green pixels, light pixels are removed first by converting the whole image to 8-bit grayscale color space. This means that each pixel has only one color value from a range of 0 to 255 representing the black and white color spectrum, respectively. Pixels with a value between 50 and 255 are then removed. This filter typically removes the largest portion of the image.

(a) (b)

**Figure 3.5: (a) Original GSV image and (b) image after filter to identify dark regions is applied.**

Rectangularity is determined by comparing the original area of the region to the area of the rectangle assigned to the region. A difference limit of 250 pixels-square was chosen as the threshold for keeping or removing regions. The geometric filters are then applied, first keeping only rectangles with an area between 15 and 400 pixels-square. Nearness to a second rectangle with a qualifying area is measure with three tests. The center of the two rectangles must be within 20 pixels, and the edge-to-edge distance must be greater than 1 pixel and less than 10 pixels. These limits were chosen in order to filter rectangles that were very distant or overlapping. Storm drains in the foreground of an image can be significantly larger than those in the background. To address this variance, edge to edge limits were calculated in addition to the overall center to center maximum limit.

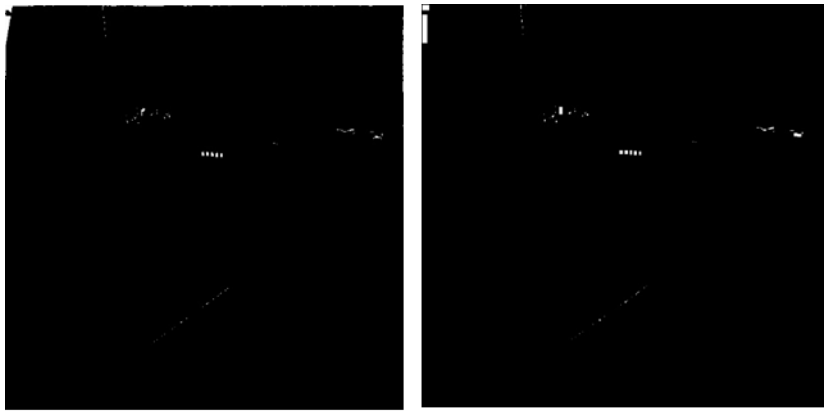



**Figure 3.6: Example of the rectangularity filter. (a) Regions remaining after green and darkness threshold and (b) rectangular areas remaining after rectangularity filter.**

A limit to filter non-parallel rectangles were imposed by comparing the angle of the long side of two qualified rectangles. The long side angle was chosen to avoid positively identifying perpendicular rectangles as storm drain attributes. This threshold was set at ±5° within parallel. The final filter is a calculation of height to width ratio. Despite the variety of perspectives that can be downloaded from GSV, drain features will always be oriented vertically within the image. A threshold of minimum 0.8 height to width ratio removes rectangles that are dramatically wider than they are tall. An example of these geometric limitations can be seen in Figure 3.7. Figure 3.8 summarizes the limits for all the filters applied during processing.

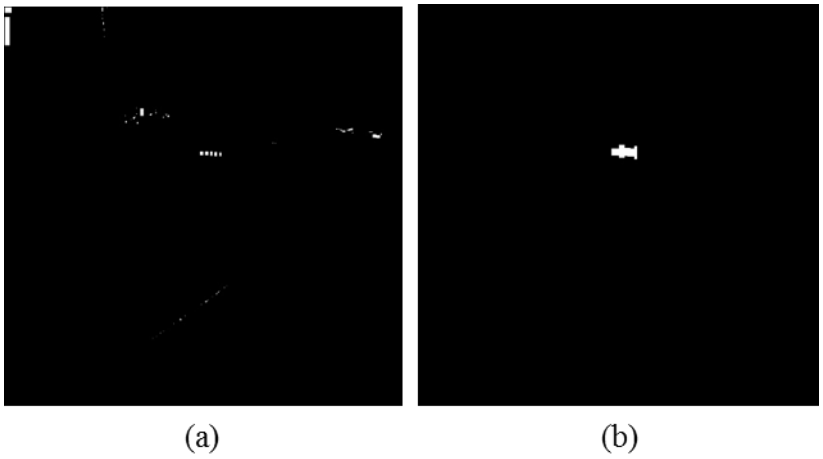



**Figure 3.7: Example of geometric filter. (a) Areas remaining after rectangularity threshold and (b) areas remaining after geometric filters.**

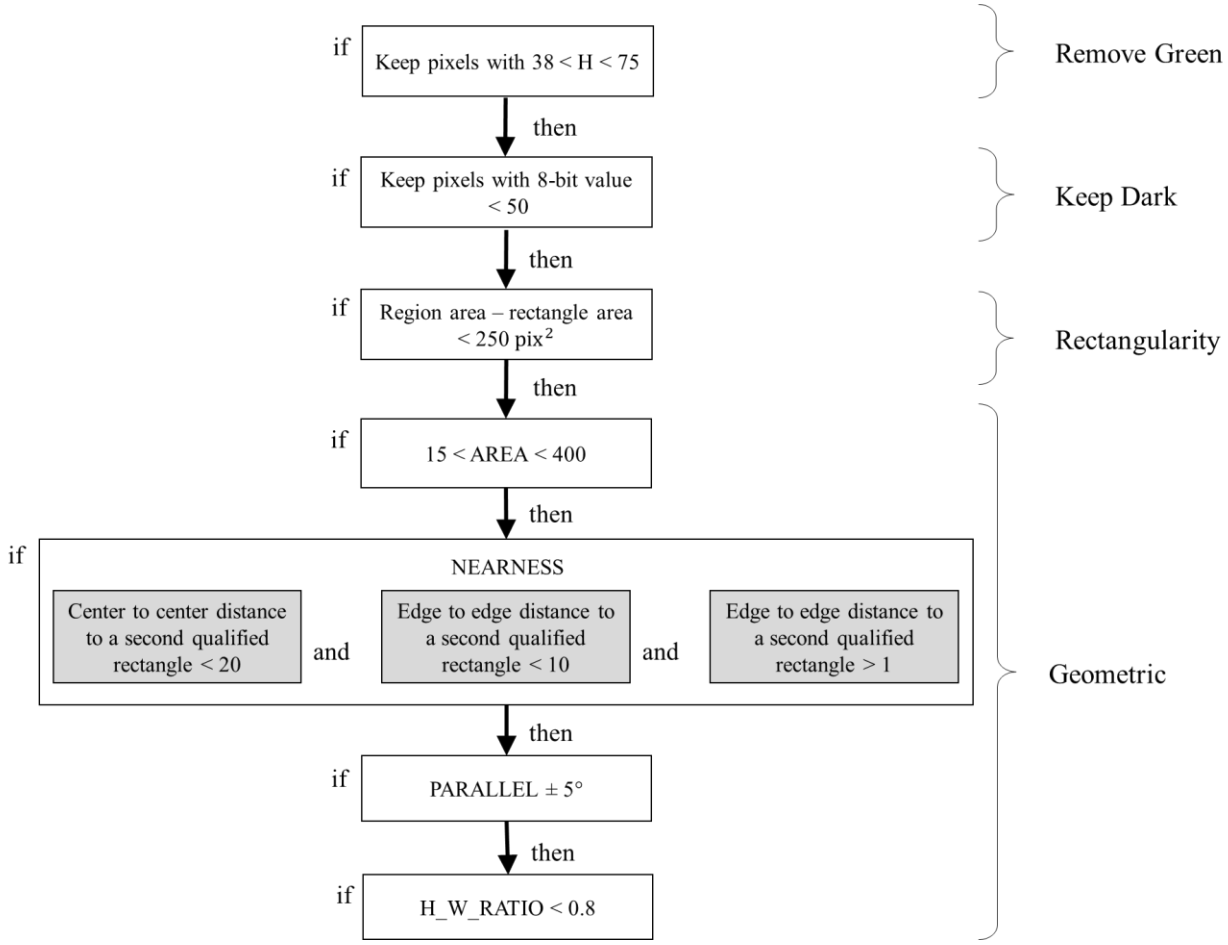



**Figure 3.8: Overview of processing filters and their threshold values.**

# CHAPTER 4: RESULTS

The following section summarizes the results found through implementing the proposed methodology. Resulting computational demand as well as accuracy of the method are reported. As mentioned, the area studied is in Champaign, Illinois. The bounding box values input to the Java GUI were [40.108327, -88.257455] by [40.109903,-88.24915]. Using an internet connection with 23.37 Mpbs download speed, 1842 images were downloaded from 307 locations in GSV within six minutes. All 1842 images were processed and the results saved in approximately twenty minutes. In addition to processing the raw GSV images, a sample of 22 images were manually pre-processed to remove non-roadway features. The following section details the results from retrieving the images as well as processing the raw and edited images.

## 4.1 Retrieved Images

The resulting images downloaded overlapped slightly, ensuring full coverage of the curb. Figure 4.1 shows an example of the six images pulled at one location along South State Street at coordinates [40.108399,-88.246757]. South State Street is a two-lane, one-way street with a bike lane and street parking on the right hand side. It can be observed from the images that the GSV vehicle collected images while driving in the right-hand lane. The width of the bike lane and the street-parking lane provide a relatively wide perspective of the curb, allowing the images to overlap only slightly. A wide perspective also prevents the line of the curb from being influenced by the dead zone located directly under the perspective of the image. The dead zone can be seen in the SW, SE, NW, and NE perspectives and limits the effectiveness of a severe pitch. While the street-parking lane provides space for a consistent collection of curb data, the presence of a vehicle utilizing on-street parking obstructs the line of sight between the camera and the curb.

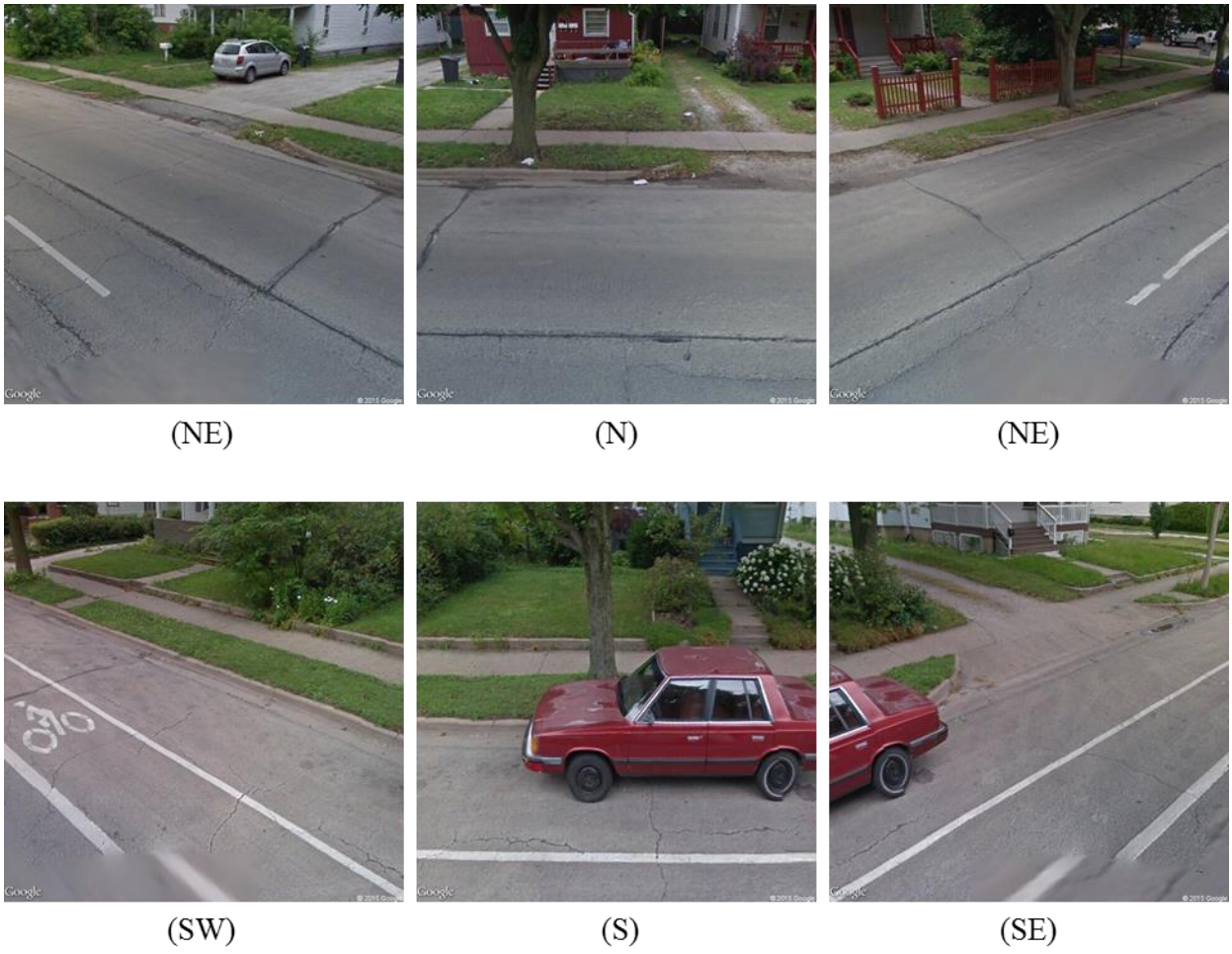



**Figure 4.1: GSV images downloaded at location [40.108399,-88.246757].**

Figure 4.2 shows an example of six images downloaded at coordinates [40.10838625,-88.2502415] along South Elm Boulevard. South Elm Boulevard is a two-way road split into two one-lane, one-way roads moving, southbound and northbound. The two one-lane roads are separated by a significant median with garden features. Due to the narrow nature of these roads, the images oriented perpendicular to the roadway do not include the curb. This exclusion creates a gap in the data gathered. If a storm drain was located in this portion of curb, it would not be possible to detect it given the image data downloaded. The dead zone is also evident in these images and is close enough to the curb to potentially interfere with the identification of a drain.

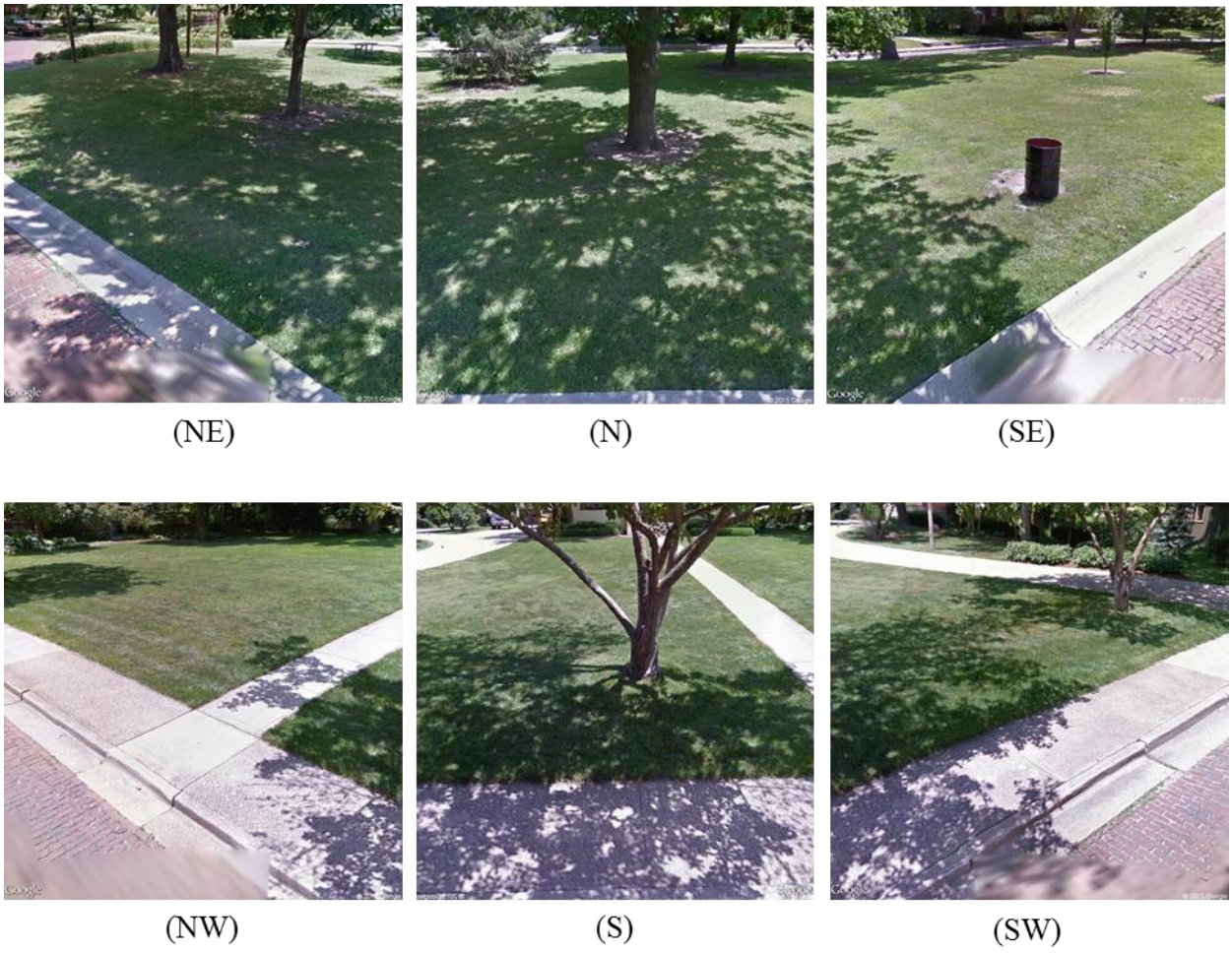



**Figure 4.2: GSV images downloaded from location [40.108399,-88.246757].**

Figure 4.3 shows images downloaded on W. John Street at coordinates [40.1090190,-88.2539880]. This location provides a sample of images downloaded from an intersection location. Unlike the straight continuous curbs found in images pulled along a road between residential blocks, all perspectives within an image contain curved curbs. It should also be noted that the perspective the images are taken from is not in the geometric center of the intersection. The variation in distance from the curb results in a change to the size of features within the image including the resolution of the storm drain characteristics. Images at this location also demonstrate the variability of environmental factors that were found within the GSV image database. The construction equipment in the southwest perspective masks any curb data for that corner.

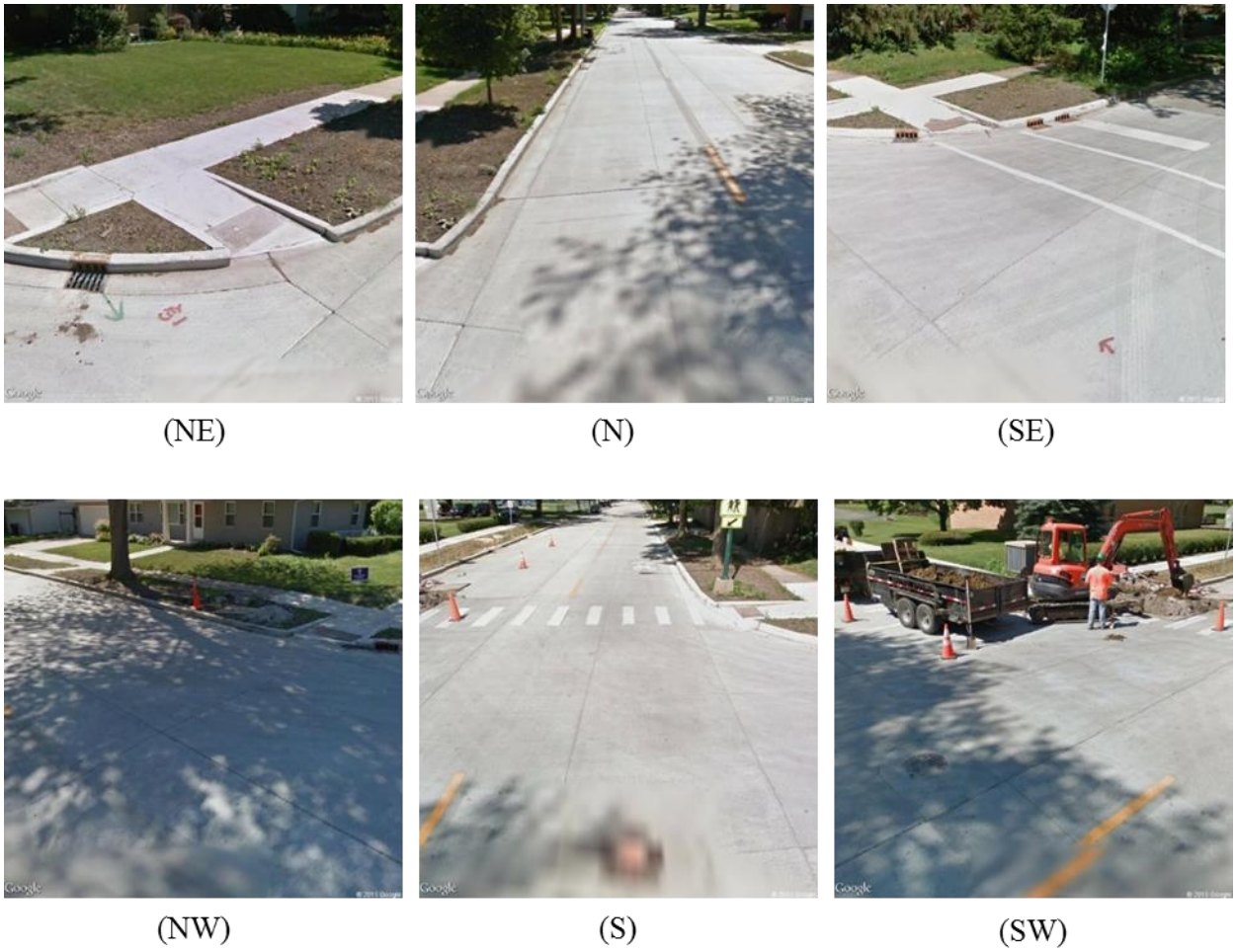



**Figure 4.3: GSV images downloaded at location [40.1090190,-88.2539880].**

## 4.2 Raw Images

This section summarizes the results found when the processing algorithm was applied to the 1842 images directly downloaded from GSV without pre-processing. Drains visible within the images are divided into two categories, perpendicular and angled. The results of processing these two type of drains are detailed. Additionally, high volume false positive detection due to non-roadway features is defined as noise and examples given.

### 4.2.1 Perpendicular Drains

While the parameters input to the Java GUI help to control the type of images downloaded, the orientation and distance of the storm drains still vary. By distance, it is meant the physical

distance of the storm drain from the location at which the GSV camera took the image. This distance influences the size of the drain and subsequently the resolution of the drain features within the image. The main differences between images are due to the orientation angle of the storm drain, distance of the drain, and lighting. Figure 4.4 and Figure 4.5 show examples of drains oriented perpendicular to the GSV camera. The drain detected in Figure 4.4 is well lit providing good contrast to the shadows found in the drain grate. Dark, shadowed regions can be seen in the foliage but because these regions have high green color values, they are identified as non-drain and removed.

(a) (b)

**Figure 4.4: GSV image containing a perpendicularly orientated storm drain in good lighting conditions. (a) Original image and (b) fully processed image.**

The perpendicular drain in Figure 4.5 is located in a moderately shadowed area. Although the entire area of the image is shadowed, the shadows do not significantly limit light allowing the drain region to remain distinctly dark. The center image in Figure 4.5 shows the processing step after green has been removed, darkness kept and the remaining regions checked for rectangularity. In this partially processed image, it can be seen that features on the home as well as features on the tree remain. The darkness and rectangularity of the window make it difficult to classify correctly. Tree bark also proved to be a source of reoccurring false positive recognition for the algorithm. In addition to its color and inherent darkness, the pattern of the bark often fits well into parallel, rectangular shapes. The final processed image positively identified the storm drain but also included false positives for the window and tree features.


**Figure 4.5: GSV image containing a perpendicularly oriented storm drain with moderate shadowing. (a) Original, (b) partially processed and (b) fully processed images.**

### 4.2.2 Angled Drains

A perpendicular perspective of a storm drain provides the clearest possible view of the geometric attributes, namely the separation between the dark rectangles created by the grate. While more sensitive to environmental factors, these distinguishable geometric attributes are still detectable at an angled perspective. Figure 4.6 shows an example of a storm drain from an angled perspective in a well-lit images. The drain is close to the location at which the image is taken making it possible to distinguish even smaller street-level grating. False detection of a portion of white fencing with vertical slats in the background should be noted.

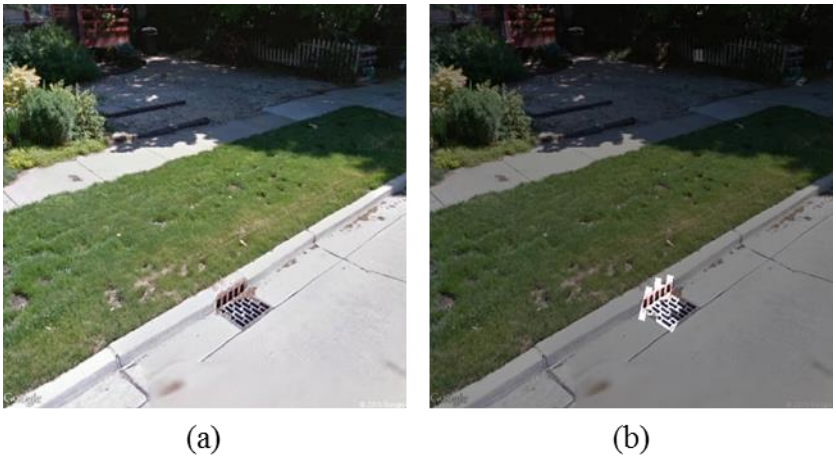



**Figure 4.6: GSV image containing a storm drain at an angled perspective in good lighting conditions. (a) Original image and (b) processed image.**

Identification of angled storm drains becomes less reliable as the angle is decreased and the distance is increased. Figure 4.7 shows a street-view image containing three grated storm drains. Two drains on the left side of the image are positively identified. The third on the right side of the image was not detected. Due to the nearly parallel angle of the grate, the drain opening on the right is represented as a single dark area and not as a series of dark rectangles. The two drains that were positively detected are at a slightly more perpendicular angle allowing the algorithm to distinguish between the grated regions and positively identify the drains.

(a) (b)

**Figure 4.7: GSV image containing multiple storm drains at angled perspectives. (a) Original image and (b) fully processed image.**

Figure 4.8 shows the original, partially processed, and final results of another angled drains. The two drains in this image were not identified by the processing algorithm. In addition to the distortion of the grating features due to angle, shadowing caused by the curb masked the unique geometric features. During processing, both drains were represented by single rectangles which do not fit the requirements found in the nearness filter.

(a) (b) (c)

**Figure 4.8: GSV image with multiple storm drains at an angled perspective. (a) Original, (b) partially processed and (b) fully processed images.**

The processing algorithm was able to detect perpendicular and angled drains within the same image. This is especially true when images are well-lit. Figure 4.9 demonstrates the processing results of an image with one perpendicular drain and two angled drains in good lighting conditions.

(a) (b)

**Figure 4.9: GSV image with storm drains at perpendicular and angled perspective in good lighting conditions. (a) Original and (b) fully processed image.**

### 4.2.3 Noise

Some examples of environmental factors which resulted in false positive identification have been previously mentioned. Below are the results highlighting the main contributors to large

volume false positive drain identification, referred to as noise. Figure 4.10 shows the original, partially and fully processed images for an intersection with two drains and a tree. The partially processed image shows the identification of the dark regions of the tree bark. As mentioned before, the darkness and directionality of tree bark proved to be a recurring issue for the algorithm.

(a) (b) (c)

**Figure 4.10: Noise caused by trees. (a) Original, (b) partially processed and (c) fully processed images.**

Another recurring source of noise in the resulting processed images was regions of shadowed dirt. Shadowed grass or bushes occasionally resulted in one or two rectangles of false positive detection, however, shadowed dirt consistently produced a high volume of noise. Figure 4.11 provides a good example of this noise. From this image it can be seen that the areas of grass were correctly negated due to their high green values. Similarly, the areas of dirt not in the shadow are negated due to their lightness. In the final processed image the shadowed dirt area produced a large number of regions identified as drain features.

(a) (b) (c)

**Figure 4.11: Noise caused by shadowed dirt. (a) Original, (b) partially processed and (c) fully processed images.**

## 4.3 Edited Images

In addition to evaluating the 1842 raw GSV images, 22 images were manually edited before the processing algorithm was applied. This was done with the understanding that algorithms have been developed for other studies which identify and remove non-road features within an image [13]. Although these methods were not directly reproducible for this study, it was seen as valuable to evaluate the processing accuracy of the present algorithm assuming that non-road features were removable.

Pre-processing of the images did not influence the positive detection of drain features, however, it significantly improved false-positive detection of non-drain features. Only one example of false-positive detection was found within the sample set. Figure 4.12 shows the original, partially processed, and fully processed image of a heavily shadowed curb. From the partially processed image it can be seen that the heavy shadows on the curb created a series of parallel rectangular shapes. Some of these rectangles were removed through the geometric filters but six rectangles remained and resulted in a final false-positive detection.

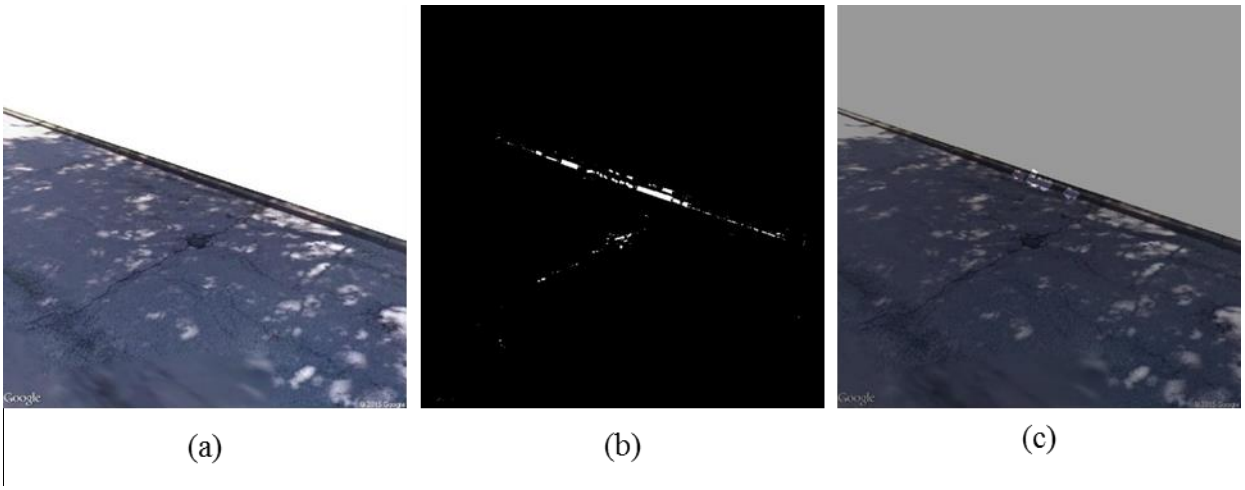



**Figure 4.12: False-positive detection of storm drain in edited images. (a) Pre-processed GSV image, (b) partially processed and (c) fully processed images.**

Thirty-four storm drains were represented within the 22 pre-processed images. After fully processing, 29 of the drains were correctly identified and 5 were not found. Figure 4.13 shows an example of a pre-processed image in which the drain was correctly identified. Figure 4.14 shows an example of a pre-processed image in which two out of the three drains were correctly identified. In both of these examples, no false-positive identification occurred.

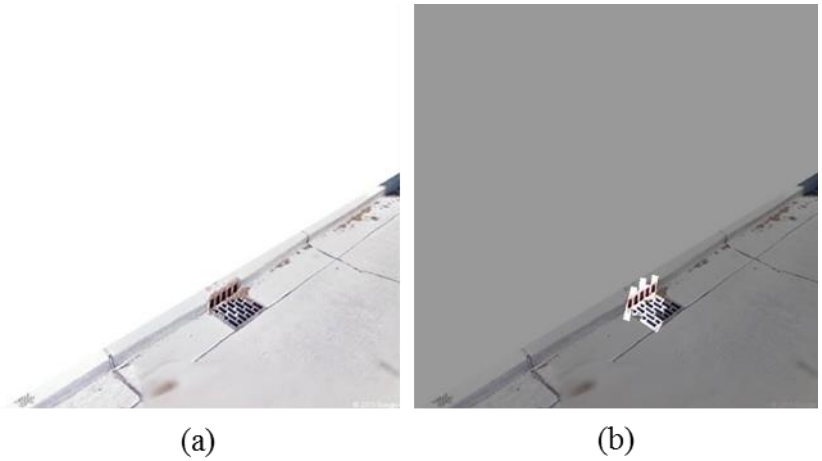
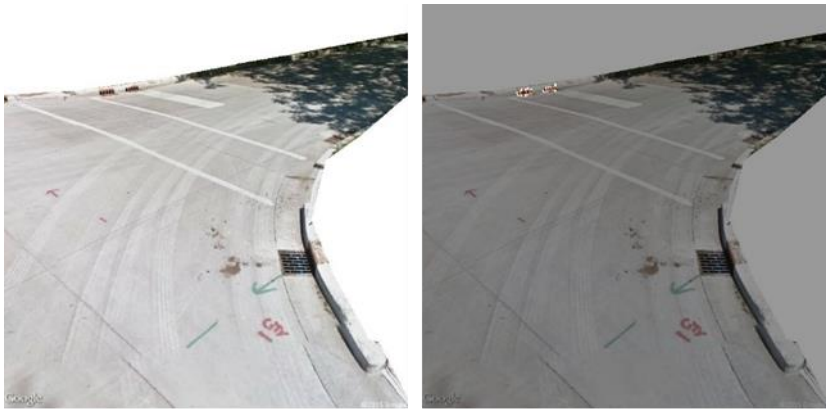



**Figure 4.13: Correct drain identification in a pre-processed image. (a) Pre-processed GSV image and (b) fully processed image.**



(a) (b)

**Figure 4.14: Two out of three correctly identified drains in a pre-processed image. (a) Pre-processed GSV image and (b) fully processed image.**

# CHAPTER 5: DISCUSSION

This chapter provides a discussion of the results presented in Chapter 4. The significance of the results in terms of the application of the overall methodology are summarized. Formative observations concerning the type of drain detected, the influence of shadows, and unique environmental features are also discussed.

## 5.1 Significance of Results

Retrieval of street-level imagery was possible using the Java GUI discussed in Section 2.1. The results from this study successfully download images for sixteen residential blocks and demonstrated the further potential to retrieve images from any large municipal areas. Limits on the application of this tool can be expected when the number of desired images exceeds the maximum value set by the Google API. While this limit may be reached for large regions of interest, it can be exceeded by purchasing a Google Maps API for Work account [4]. Possible limitations could also apply when downloading images in rural areas that may not by covered within the GSV database, however, this is not expected as GSV data is quite extensive.

The usefulness of the images downloaded was a function of the perspective parameters input to the Java GUI tool, as well as environmental factors related to the location. Pitch, field of view, and heading were adjusted to retrieve the best images for a residential, two-lane road. These parameters could be changed to improve resulting image perspective for different types of roadways. Environmental factors such as shadowing and line of sight obstruction were found to be unavoidable and could not be improved upon when using GSV images.

Processing of the retrieved GSV images identified a majority of the storm drains with grated openings. The overall accuracy of the method was influenced by line of sight obstructions, image perspective of drain features and noise. Line of sight obstructions prevented storm drains from being represented within the image resulting in missed drain locations. Examples of obstructions are vehicles, construction work and debris in the drain itself. Image perspective influenced the accuracy of the algorithm but did not entirely prevent drain features from being detected. Drains

oriented perpendicular to the camera perspective were more reliably detected than those parallel. False positive drain identification was caused by noise within the images. Noise was significantly reduced by removing non-road features. While line of sight obstructions are unavoidable, parallel drain identification and noise reduction are areas in which the algorithm could be improved.

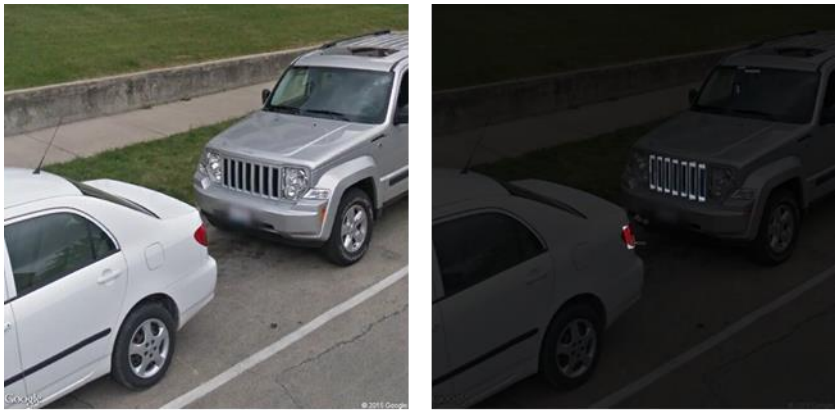
## 5.2 Formative Observations

Non-grated inlets proved to be significantly more difficult to detect than grated inlets. Only limited amount of data reduction was possible based on color features within the GSV images. This left a significant portion of the image data to be filtered based on shape and clustering techniques. The results show that much of the data remaining after the color filters were applied was rectangular in nature and often represented human made features. This made it impossible to distinguish a single rectangle representing a drain from a single rectangle representing a non-drain feature. Because of this difficulty, the scope of this project was limited to designing an algorithm which could identify grated features.

Shadows were also difficult to distinguish. The results show that shadowing reduces the reliability of the algorithm but does not fully inhibit detection. It was seen that shadows caused by distant obstructions, such as tall trees which reduce sunlight on the roadway, were less disruptive than direct obstructions, such as curb self-shadowing or ground-level foliage. This is reasonable because, while they may cover more area, distant obstruction create less severe shadowing and allow more light on the roadway. This observation has implications for further application of the algorithm. It can be expected that if the area of interest for storm drain mapping is in an area with tall trees there may be some disruption. On the other hand, if the area of interest contains a high volume of ground level features which shadow the roadway, the accuracy should be expected to be severely influenced.

Ground level features within the natural roadside environment often contain patterns and color characteristics which are remarkably similar to storm drains. While the high volume, false detection of noise can be improved upon through enhanced shape and pattern based filters, some features found in the natural environment will require more extensive improvements to the

algorithm. Figure 5.1 shows an example of the detection of a car grill. The grill consists of a series of dark, parallel, rectangular areas which are falsely identified as a storm drain.



(a) (b)

**Figure 5.1: False positive drain identification caused by car grill features. (a) Original GSV image and (b) processed image.**

# CHAPTER 6: CONCLUSION

## 6.1 Summary

Stormwater management systems are designed to protect the property, health, and safety of communities all over the world. These systems are highly dependent on the location and functionality of the street-level storm drain network. Knowledge of the locations of storm drains is essential for understanding and improving the overall performance of modern runoff and waste water management; however, due to the complexity of the systems, the frequency of updates and the accuracy of the data can be issues.

This study found that the use of street-level imagery was a practical method for identifying and extracting storm drain information. Despite limitations of available image resolution and potential large volume downloads, the publicly available GSV images provided consistent coverage of the curbside with few gaps. Gaps in curb data were found as a function of line of sight obstructions (parked cars, etc.) and perspective limitations (one-lane roads with no shoulder), and not as a function of GSV image availability. These factors should be considered when applying this method to other regions of interest.

Processing of GSV images was found to be insufficient using only color and shape based filters. While storm drain detection within the images was successful using this methodology, false identification of features not representative of drains was extensive. After applying a manual pre-processing step removing non-road features from a small sample of images, the color and shape based processing algorithm successfully identified 29 out of 34 storm drains and only one false positive. This tool demonstrates the conceptual and computational viability of using street-level images as a novel method for mining and mapping of storm drain locations.

## 6.2 Future Work

Future work for this project should be focused on improving the accuracy and detail with which data is extracted from the images and reported to the user. The processing algorithm detailed in the methodology could be improved by applying a pre-processing step which filters out non-road features. Because of the high volume of noise caused by non-road features it was not practical to include single opening drains and horizontal man-holes to the processing goals. However, once non-road features are removed further work could be done to search for these alternative components of the runoff management system.

The detail of the coordinate data exported from the processing step is another point of future improvement. Each image analyzed through the current methodology results in one and only one exported coordinate point which is related to the coordinate point at which the image was taken and the heading perspective parameter. The exported coordinates could be improved upon by identifying the distance of the drain from the point at which the image was taken. By assigning location details to a cluster of identified features within the image rather than the image as a whole, further improvement could be done to export multiple coordinates for a single image if multiple drains were present

# REFERENCES


[1] Minnesota Department of Transportation, "Drainage Manual: Chapter 8 Storm Drainage Systems," 2000.

[2] EPA, "Report to Congress: Impacts and Control of CSOs and SSOs," EPA, Washington, DC, 2004.

[3] D. Anguelov, C. Dulong, D. Filip, C. Frueh, S. Lafon, R. Lyon, A. Ogale, L. Vincent, J. Weaver and Google, "Google Street View: Capturing the World At Street Level," *IEEE Computer Society*, 2010.

[4] Google, Inc., "Google Developers: Google Street View API," 28 May 2015. [Online]. Available: https://developers.google.com/maps/documentation/streetview/. [Accessed 29 May 2015].

[5] A. Rundle, M. Bader, C. Richards, K. Neckerman and J. Teitler, "Using Google Street View to Audit Neighborhood Environments," *American Journal of Preventive Medicine*, pp. 94-100, 2011.

[6] J. Xiao, T. Fang, P. Tan, P. Zhao, E. Ofek and L. Quan, "Image-based Facade Modeling," 2008.

[7] J. Xiao, T. Fang, P. Zhao, M. Lhuilier and L. Quan, "Image-based street-side city modeling," in *ACM Transactions on Graphics*, Yokohama, Japan, 2009.

[8] V. Balali, E. Depwe and M. Golparvar-Fard, "Multi-class Traffic Sign Detection and Classification Using Google Street View Images," in *Information Systems and Technology (ABJ50)*, Washington, DC, 2014.

[9] K. Hara, J. Sun, R. Moore, D. Jacobs and J. E. Froehlich, "Tohme: Detecting Curb Ramps in Google Street View Using Crowdsourcing, Computer Vision, and Machine Learning," 2014.

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

[11] I. Abdel-Qader, O. Abudayyeh and M. Kelly, "Analysis of Edge-Detection Techniques for Crack Identification in Bridges," *Journal of Computing in Civil Engineering* , pp. 255-263, 2003.

[12] D. Hoiem, A. A. Efros and M. Hebert, "Recovering Surface Layout from an Image," *International Journal of Computer Vision,* vol. 75, no. 1, pp. 151-172, 2007.

[13] D. Hoiem, A. A. Efros and M. Hebert, "Geometric Context from a Single Image," in *Proceedings of the Tenth IEEE International Conference on Computer Vision (ICCV'05),* 2005.

[14] D. Hoiem, A. A. Efros and M. Hebert, "Automatic Pop Up," Robotics Institue, Carnegie Mellon University, Pittsburgh, 2005.

[15] J. D. Fulton, "Autonomous Golf Cart Vision Using HSV and Commercial Webcam," Electrical Engineering Department California Polytechnic State University , San Luis Obispo, CA, 2011.

[16] "OpenStreetMap," 1 May 2015. [Online]. Available: http://wiki.openstreetmap.org/wiki/Downloading_data. [Accessed 3 May 2015].

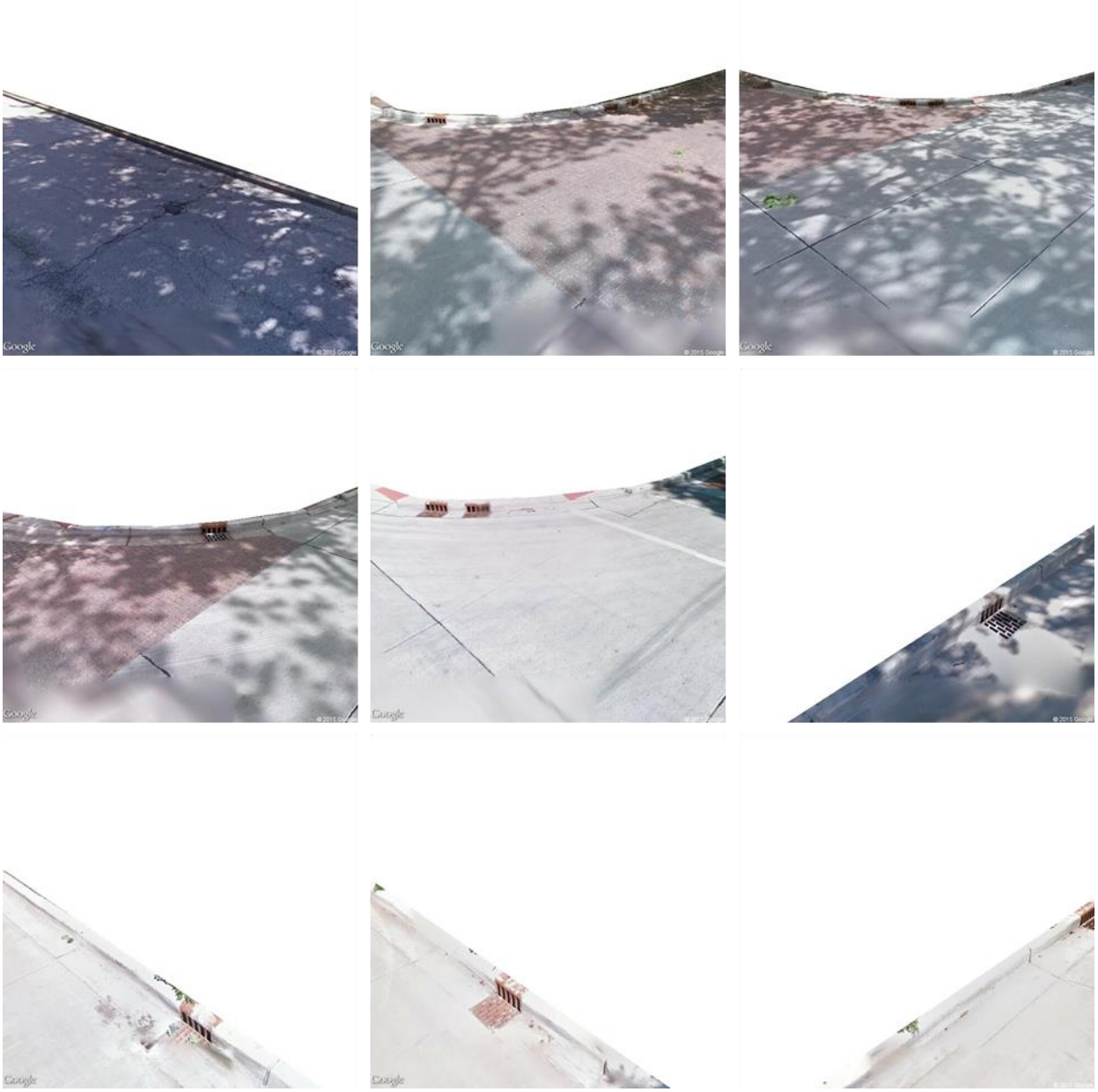
[17] D. Jordan, "Implementation Benefits of C++ Language Mechanisms," *Communications of the ACM,* vol. 33, no. 9, pp. 61-64, Sept 1990.

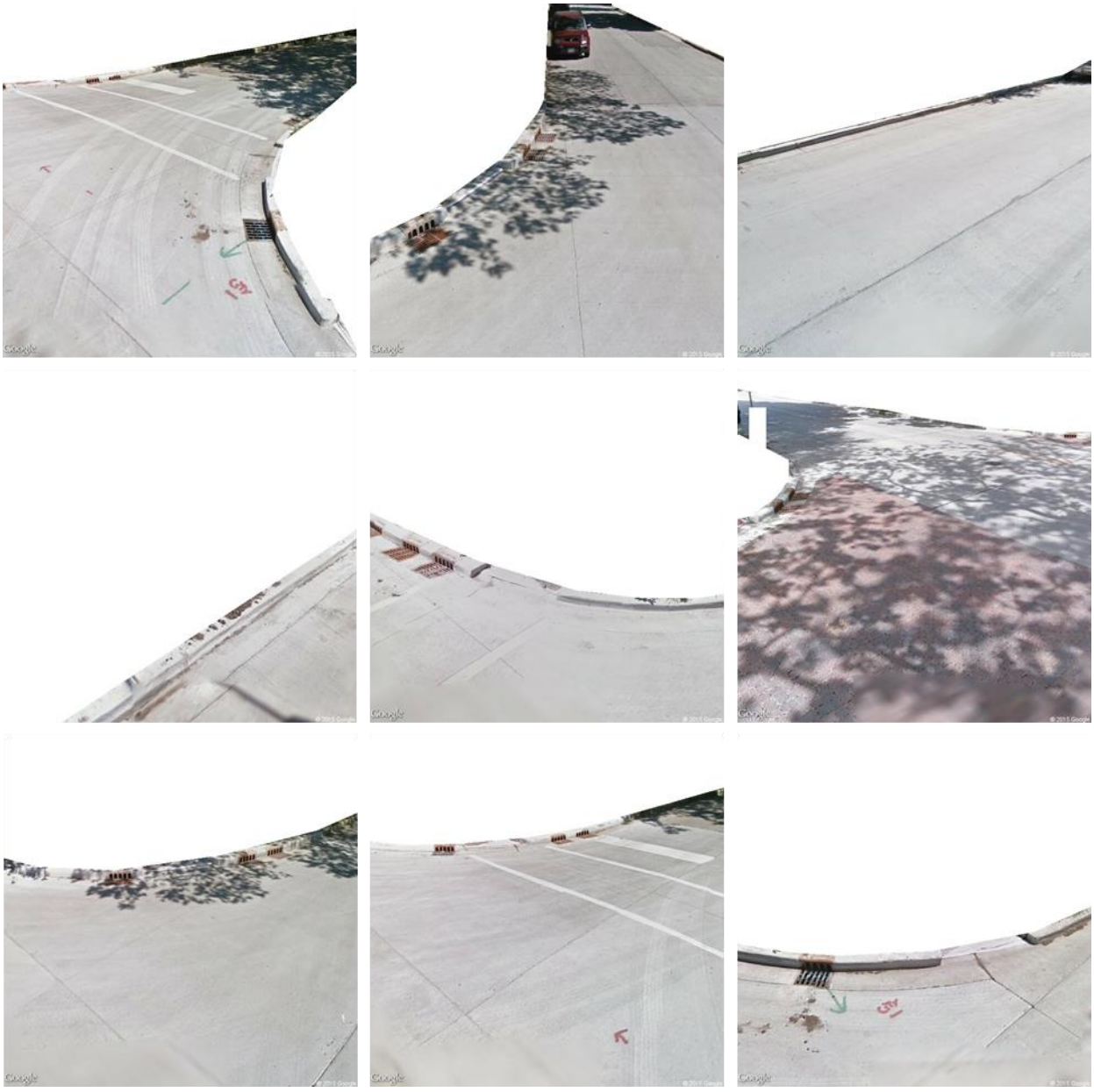
[18] OpenCV, "Miscellaneous Image Transformations: cvtColor," 24 Feb 2015. [Online]. Available: http://docs.opencv.org/modules/imgproc/doc/miscellaneous_transformations.html?highlight=cvtcolor. [Accessed 29 May 2015].

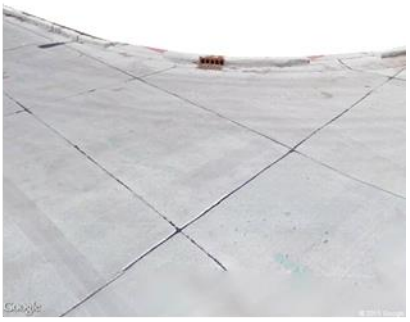
[19] V. Mistry and R. Makwana, "Survey: Vision based Road Detection Techniques," *International Journal of Computer Science and Information Technologies,* pp. Vol. 5 (3) 4741-4747, 2014.

[20] A. Rai, "Designing Green Stormwater Infrastructure for Hydrologic and Human Benefits: An Image Based Machine Learning Approach," University of Illinois, Urbana, Illinois, 2013.

# APPENDIX A: 22 PRE-PROCESSED IMAGES







# APPENDIX B: SOURCE CODE FOR GSV IMAGE RETRIEVAL

The Java source code for downloading images from Google Street View within a user defined bounding box may be found in the supplemental folder named **Drain_Detect.**

# APPENDIX C: SOURCE CODE FOR PROCESSING IMAGES

The source code, written in C++, which processes images and saves a csv file with the coordinates of identified storm drains may be found in the supplemental file named **hello_opencv.**