C2SMART
CONNECTED CITIES WITH SMART TRANSPORTATION
A USDOT University Transportation Center

New York University
Rutgers University
University of Washington
University of Texas at El Paso
The City College of New York

Integrated Analytics and Visualization for Multi-modality Transportation Data

April 2019

C2SMART Center is a USDOT Tier 1 University Transportation Center taking on some of today's most pressing urban mobility challenges. Using cities as living laboratories, the center examines transportation problems and field tests novel solutions that draw on unprecedented recent advances in communication and smart technologies. Its research activities are focused on three key areas: Urban Mobility and Connected Citizens; Urban Analytics for Smart Cities; and Resilient, Secure and Smart Transportation Infrastructure.

Some of the key areas C2SMART is focusing on include:

Disruptive Technologies

We are developing innovative solutions that focus on emerging disruptive technologies and their impacts on transportation systems. Our aim is to accelerate technology transfer from the research phase to the real world.

Unconventional Big Data Applications

C2SMART is working to make it possible to safely share data from field tests and non-traditional sensing technologies so that decision-makers can address a wide range of urban mobility problems with the best information available to them.

Impactful Engagement

The center aims to overcome institutional barriers to innovation and hear and meet the needs of city and state stakeholders, including government agencies, policy makers, the private sector, non-profit organizations, and entrepreneurs.

Forward-thinking Training and Development

As an academic institution, we are dedicated to training the workforce of tomorrow to deal with new mobility problems in ways that are not covered in existing transportation curricula.

Led by the New York University Tandon School of Engineering, C2SMART is a consortium of five leading research universities, including Rutgers University, University of Washington, the University of Texas at El Paso, and The City College of New York.

c2smart.engineering.nyu.edu

Integrated Analytics and Visualization for Multi-modality Transportation Data

Claudio T. Silva
New York University

Juliana Freire
New York University

Fabio Miranda
New York University

Marcos Lage
Universidade Federal Fluminense

Harish Doraiswamy
New York University

Maryam Hosseini
Rutgers University

Eric Tokuda
Universidade de São Paulo

Gabriel Ferreira
Universidade de São Paulo

Roberto M. Cesar-Jr.
Universidade de São Paulo

Disclaimer

The contents of this report reflect the views of the authors, who are responsible for the facts and the accuracy of the information presented herein. This document is disseminated in the interest of information exchange. The report is funded, partially or entirely, by a grant from the U.S. Department of Transportation's University Transportation Centers Program. However, the U.S. Government assumes no liability for the contents or use thereof.

Acknowledgements

We thank Carmera for their collaboration. This work was supported in part by: NSF awards CNS-1229185, CCF-1533564, CNS-1544753, CNS-1730396, CNS-1828576; CNPq; FAPESP; FAPERJ; the Moore-Sloan Data Science Environment at NYU; and C2SMART. C. T. Silva and J. Freire are partially supported by the DARPA D3M program. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of DARPA.

Executive Summary

Urban informatics is an important field that is attracting increasing attention from academia, government and industry. One particular area of interest in this field is in modeling movement of people and goods around cities. A city model should represent information that allows the derivation of analytical methods to answer questions like: How do cities evolve? How can cities be compared, clustered, and distinguished?

This research project aims to develop a data-driven approach for modeling cities, which plays a fundamental role in urban planning. This requires the identification of areas in the city that have similar characteristics, such as similar urban fabric, building facade, or even a specific item of interest, such as broken curbs or pedestrians. However, new data sets consisting of dense collections of images are now becoming available, which can significantly help in answering many of the above questions. We leverage a new data set composed of tens of millions of images from New York City captured over a period of a year by cameras mounted on top of cars and produced by Brooklyn-based start-up Carmera. This data, by providing comprehensive coverage of not only the various streets of the city but also different time periods, has the potential to provide users with a visual perspective of the city that was not possible before.

In this project, we first present a framework that leverages recent advances in computer vision to efficiently handle such a large collection of complex images. We then propose to construct a spatiotemporal map of relative pedestrian density. Due to the limitations of state-of-the-art computer vision methods, such automatic detection of pedestrians is inherently subject to errors. We model these errors as a probabilistic process, for which we provide theoretical analysis. Through numerical simulations, we demonstrate that, within our assumptions, our methodology can supply a reasonable estimate of pedestrian densities and provide theoretical bounds for the resulting error. Lastly, we present an interactive visual analysis tool for the exploration of this large collection of images. Our approach computes a set of feature vectors for each image in this large street-level collection and makes use of a memory-efficient index to interactively answer queries about this data. We then design a visual interface that couples the image data with other urban data sets and allows users to interactively query, explore and analyze visuals in a city over both space and time. This provides them with the opportunity to not only virtually audit the built environment but also help answer numerous questions about the complex system of cities. Working in collaboration with urban planning researchers, we illustrate the utility of our framework through several use cases.

Table of Contents

List of Figures

Street-level Images

The data-providing company has been equipping vehicles in NYC with a set of four inexpensive mobile phones, each one facing a particular direction (front, back, left, right). As the cars travel the boroughs of NYC, the cameras from the mobile phones capture images at regular time intervals. Every image in the data is accompanied by the following metadata: location captured by the mobile phone (latitude and longitude) of the car when the image was taken, time and camera orientation.

It is important to note that, unlike in Google Street View (GSV), where cars were deployed specifically to capture street-level images, the cars used for capturing images are regular vehicles. Due to this, there is no control over the quality of the images such as illumination, weather, traffic condition or blurriness (due to vehicular speed).

On the other hand, given the inexpensive image capturing approach, these images are spatially and temporally denser than other data sets, such as GSV or Bing Streetside.

Framework Description

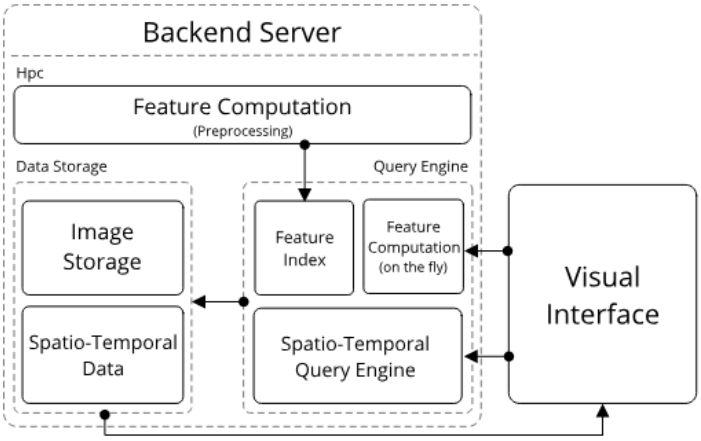

Figure 1: System architecture.

Our system can be broadly divided into two parts as illustrated in Figure 1: the visual interface (see Section 5.3), and a backend server. The backend server is responsible for enabling real-time responses to queries composed using the visual interface. It is composed of a data storage module and a query engine. The images are simply stored as individual files on disk. As mentioned earlier, one of the goals of this system is to also support other urban data sets, which are primarily spatiotemporal. Note that the metadata corresponding to the images is also spatiotemporal. We use MonetDBLite[1], a lightweight column-oriented embedded analytical SQL database (code available at

https://github.com/MonetDB/MonetDBLite-C), as a data store and store the spatio-temporal data sets as tables. The advantage of using a column store is that queries involving only a subset of the data columns are more efficient than traditional row-oriented approaches. As we describe later in Section 5.3, the following types of queries can be issued by the interface:

1. spatiotemporal selection queries.
2. spatiotemporal aggregation queries.
3. image similarity queries.

We built a query engine to support such queries. Spatial selection queries are handled by first performing a coarse query from the data store that satisfies the time period constraint. The points resulting from this coarse query are then tested in parallel for the spatial constraint and returned to the interface.

Handling spatiotemporal aggregation queries is more complicated, and we chose to use the fast GPU-based RasterJoin[2] algorithm (code available at https://github.com/VIDA-NYU/raster-join) for this purpose. Given a spatial aggregation query, as before we first run a coarse query to filter for the time period, and the result from this is fed to the RasterJoin algorithm to compute the required spatial aggregation.

To handle image similarity queries, in a preprocessing step, we first compute a feature index that computes and stores the semantic features of the images. This index is then used to perform similarity queries.

Pedestrian Density Estimation

Introduction

Pedestrians are an integral and pervasive aspect of the urban environment. Real estate, consumer patterns, public safety, and other aspects of city life are deeply intertwined with the variations of pedestrian densities across a city. However, current methods for estimating the distribution of people within a city tend to be expensive and mostly produce a space sampling of a few locations.

In this chapter, we examine a new method to obtain an estimation of pedestrian density. We utilize recent advances in computer vision to find people within previously intractable collections of images to compile a relative density map.

In order to take into account the errors inherent to visual objects detection, we model it as a *probabilistic detection*. Using our model, we provide a closed form and bounds for the asymptotic error

of the sampling process. We compare these formulas to numerical simulations of the sensing process. Our results suggest that computer vision produces usable data, despite the inherent noise.

To test our method, we utilized over 40 million street-level images provided by Carmera. The images were obtained using special fleets that traveled through the region of Manhattan in New York City over the course of the year. A sample of this data was used to benchmark several state-of-the-art computer vision algorithms. We then utilized the top performing algorithm in a case study to map pedestrian densities in Manhattan.

In this chapter, we make the following contributions:

1. A new method for the analysis of the spatial variation of urban pedestrians densities utilizing state of the art, but imperfect, computer vision algorithms.
2. A closed form function and bounds for the asymptotic error of the resulting pedestrian densities.
3. The results of simulations validating the sampling process and the derived asymptotic error.
4. A benchmark of several detection algorithms, along with the variation in their parameters, for the purpose of pedestrian detection.
5. A case study demonstrating the resulting densities for a collection of images from the City of New York.

Related Work

There are many ongoing efforts on the use of urban data to achieve citizen-centered improvements [3]. Governments and organizations in urban environments collect a vast amount of data daily, [4] amassing a large assortment of information about mobility, crime, pollution, and more. The collection and use of this information has been attracting attention from academics, governments and corporations [5]. The work from Arietta et al. [6] explores the correlation of visual appearance of pictures and the attributes of the region to which it pertains. They collected images from Google Maps [7] and indicators from multiple regions and trained a model [8] to predict the indicator based on images. The city attributes include violent crime rates, theft rates, housing prices, population density and trees presence. Results show that the visual data can be efficiently used to predict the region's attributes. Additionally, the regressor trained in one region showed reasonable results when tested in a different city.

A pedestrian map of the city has numerous applications for urban planners, including the design of public transport networks and of public spaces [9]. One approach to obtain a citywide count of pedestrians is to have people scattered around the city manually counting the pedestrians nearby. This approach is laborious, requiring significant manpower and time to perform the measures. Another possibility explored in [10] is to use cellphone use data to perform the pedestrian count. Two clear

limitations of this approach are that these data are not public, and the coverage is restricted to the places where the carrier signal is present. Additionally, it is hard to know whether the cell signal is from a pedestrian or from someone in a building or from someone in a car.

Alternatively, we can consider the visual task of finding the pedestrians in city images. One approach to this challenge consists of using the histogram of oriented gradients as the features vector and a support vector machines for the classification task [11]. In the context of deep neural networks [12] [13], the work of [14] introduced an approach that tries to solve this task by using a unified network that performs region proposal and classification. In this way, the method accepts annotations of multiple sized objects during the training step, and during the testing stage, it performs classification of those objects in images of arbitrary sizes. In [15] the authors follow the two-stage region proposal and classification framework of [14] and propose the Region-based Fully Convolutional Networks, (R-FCN) which incorporate the idea of position-sensitive score maps to reduce the computational burden by sharing the per-RoI computation. Such speed alterations allow the incorporation of classification backbones such as [16].

There are several city images repositories that considers pedestrians, some of them obtained using static cameras [17–19] and others obtained using dynamic ones [20–22]. Such configuration of sensors has long been studied in the *sensor network* field[23–25] and an important aspect of these networks is whether the sensors are static or mobile. In [26] the authors explore the setting of a network composed of both static sensors and of mobile sensors. The holes in the coverage of the static sensor network are identified and the mobile sensors are used to cover the holes. A common problem in sensor networks is the *k-coverage problem* defined in [27], that aims to find the optimal setting of sensors such that any region is covered at least by k sensors. In [28] the authors perform the task of counting people based on images obtained through a wireless network of static sensors.

Apart from controllable mobile sensor networks, many works explore data collected from collaborative uncontrolled sensors [29] such as from vehicle GPS [30,31], mobile phone sensors [32–34] and even from on-body sensors [35].

The work of [36] considers the problem of using GPS data from a network of uncontrolled sensors to reconstruct the traffic in a city. They do that in two steps: initial traffic reconstruction and dynamic data completion. This approach allowed the authors to get a complete traffic map and a 2D visualization of the traffic.

There are many ways to model the movement of mobile nodes in a sensor network, the so-called *mobility models* [37]. A simple one is the *random walk mobility model* [38] where at each instant in time each particles gets a direction and a speed to move. In the *random waypoint mobility mode* [39], in turn, particles are given destinations and speeds. They travel toward their goal and once they arrive at the destination a new goal and speed are given. The *Gauss-Markov mobility model* [40] attempts to eliminate

abrupt stops and sharp turns present in the random waypoint mobility model. It is done by computing the current position based on the previous position, speed and direction.

Simulation of wireless sensor networks has long been studied [41–43] because it allows a complete analysis of system architectures by providing a controlled environment for the system [44]. The real-life systems non-determinism is simulated using pseudo random number generators [45]. Among the large number of pseudo random number generators [46], a popular algorithm is the Mersenne Twister [47] due to its efficiency and robustness.

Pedestrians and Sensors Flow Model

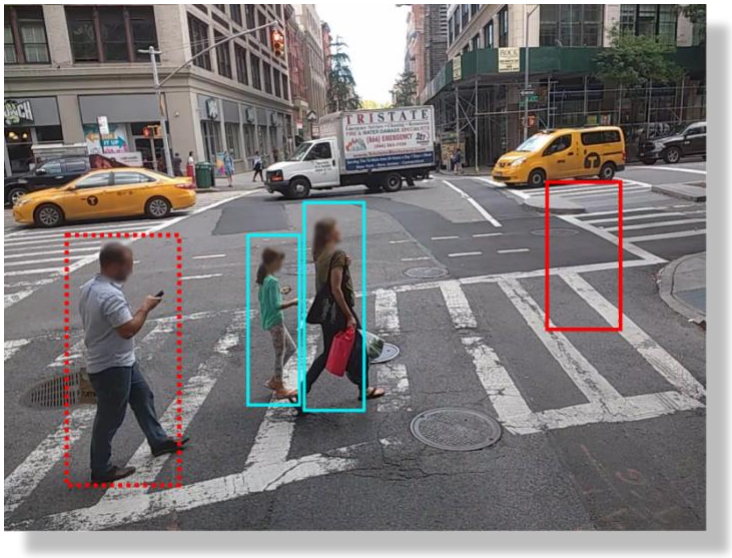

Figure 2: A hypothetical illustration of the type of detection errors considered in this paper.

As current pedestrian detection algorithms are far from perfect, it is natural to wonder about the accuracy of any pedestrian count resulting from their use. As shown in Figure 2, a number of detection errors can occur. The person on the left, outlined with the red dotted line, was not identified by the detector and is a false-negative. The rightmost detection, showing an empty red box, is a false-positive. The two correct detections in the center are true-positives. Missing from this example are true-negatives, which are not a useful concept in this situation due to the overwhelming number. In this section, we provide a theoretical analysis of the effect of algorithmic errors on the final count.

In our model, we assume that the world is modeled by a number of small regions, or buckets, each of which we intend to measure a density for. Sensors and people move around a world in some random fashion. At regular intervals, each sensor takes an independent measurement of the nearby pedestrian count and updates the recorded density at its current location, x. More formally, each time a sensor

takes a sample, it obtains a measurement represented by the random variable $\mathrm{N}(\mathrm{x})$. While we don't specify the distribution of $N(x)$, we assume that the expected value follows the formula

$$E[N_i(x)] = pn_i(x) + \lambda$$

Here $\mathrm{n_i}(\mathrm{x})$ is the actual number of people in the location and time being sensed. p is a number giving the success rate of the vision algorithm and λ indicating its false positive rate.

The result of this process is the density of people at each location, $\psi(\mathrm{x})$.

$$\psi(x) = \frac{1}{k}\sum_i N_i(x)$$

For comparison, the ground truth density $\phi(x)$, defined respectively by (where k is the number of steps and samples),

$$\phi(x) = \frac{1}{k}\sum_i n_i(x)$$

The expected value of $\psi(x)$ is

$$E[\psi(x)] = p\phi(x) + \lambda$$

In other words, $\psi(x)$ is a biased estimator of $\phi(x)$. Unless our sensing algorithm precisely follows the previous equation, we are unable to transform this biased estimator into an unbiased one. Furthermore, even in the ideal case, p and λ may not be known. Instead, we directly utilize $\psi(x)$ and attempt to find a relative histogram. That is, we expect to get a number proportional to the density of the number of people at a location and not the actual density. As such, for any constant a, our density is equivalent to one scaled to $\psi'(x) = a\psi(x)$. Treating the distribution as a vector, we measure the direction but not the magnitude. In the terminology of group theory, our measurement suggests a density within the equivalent class:

$$[\psi] = \{\, a \in R_+ \mid a\psi \,\}$$

To validate our measurement we need a metric that indicates how well the equivalent class compares to the ground truth distribution $\phi(x)$. To do that, we compare the ground truth to the unique closest element within the equivalent class. As a vector projection, this minimum element is:

$$\psi' = \psi\frac{<\psi, \phi>}{|\psi|^2} if\ |\psi| \neq 0$$

$$\psi' = 0\ if\ |\psi| = 0$$

which we can then compare using the usual Euclidean metric $|\psi' - \phi|$. However, this metric depends on the number of locations in the map, as well as the number of people. As such, we normalize the metric to between 0 and 1, to obtain a final metric:

$$\frac{|\psi' - \phi|}{|\psi'| + |\phi|}$$

Over long periods of time, we expect the asymptotic error to approach:

$$\frac{|\psi' - \phi|}{|\psi'| + |\phi|} \to \frac{\lambda}{h} \frac{\sqrt{c^2\left(p + \frac{\lambda}{h}\right)^2 + \left(pc^2 + \frac{\lambda}{h}\right)^2 - 2\left(p + \frac{\lambda}{h}\right)\left(pc^2 + \frac{\lambda}{h}\right)}}{\left(pc^2 + \frac{\lambda}{h}\right)\sqrt{p^2c^2 + \left(\frac{\lambda}{h}\right)^2 + 2p\frac{\lambda}{h}} + \left(p^2c^2 + 2p\frac{\lambda}{h} + \left(\frac{\lambda}{h}\right)^2\right)c}$$

Here $h > 0$ is the average density of people and $c \geq 1$ describes the distribution of ϕ. However, c can best be thought of as parameters that describe the asymptotic error. Both of these parameters depend on the resolution of the heat map in addition to pedestrian distribution. In many cases c can not be determined, as such we can use the inequality:

$$\lim_{k \to \infty} \quad \frac{|\psi' - \phi|}{|\psi'| + |\phi|} \leq \frac{\sqrt{c^2 - 1}\lambda}{2c^2 hp} \leq \frac{1}{4}\frac{\lambda}{hp}$$

It is important to note that h needs to be the ground truth density of people, in the same units of ϕ. If only the sampled average density, $\hat{h}$, is know, the unbiased estimator of h, $\frac{\hat{h} - \lambda}{p}$ can be used. This leads to the bounds

$$\frac{1}{4}\frac{\lambda}{hp} \approx \frac{1}{4}\frac{\lambda}{\bar{h} - \lambda}$$

This final formula is only dependent on the false positive rate of the sensing algorithm and the average density of sensed objects measured by process, making it suitable for practical sensing applications. This function is only useful when $\lambda \leq \bar{h}$. In that domain, it is a monotonically increasing function of λ. Thus, if $\lambda$is not precisely known, it is best to err on the side of larger values.

Simulation

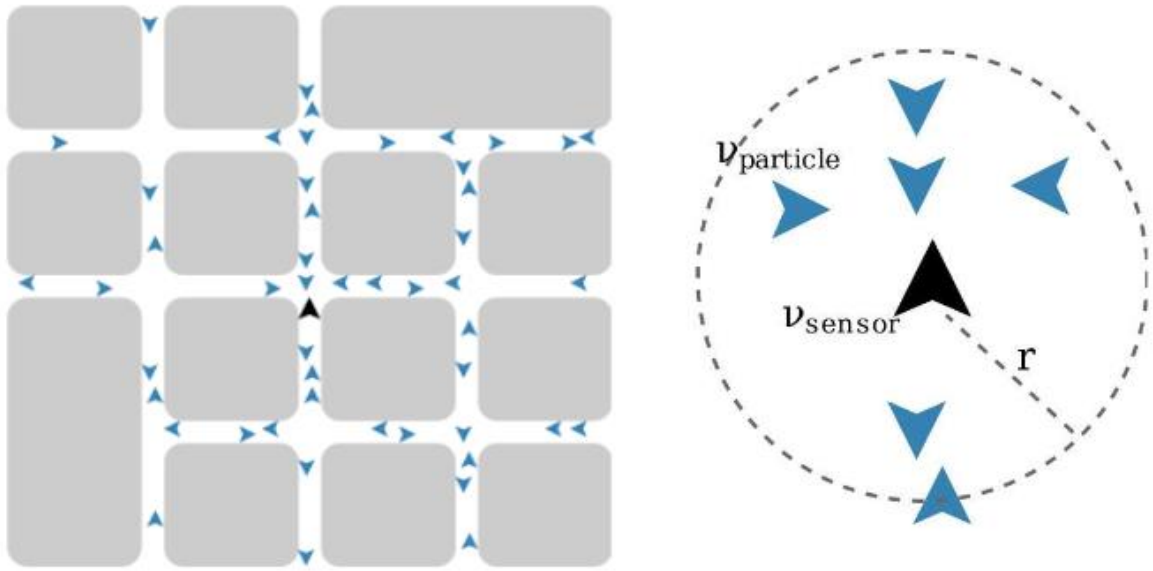

Figure 3: (Left) An illustration of our simulation containing a sensor (center) moving through an environment with numerous pedestrians. (Right) Each sensor moves with uniform speed v_s and is able to sense people within a radius of r. Each sensing operation has a probability p of correctly detecting each person and, on expectation, finds λ false positives. People move with uniform speed v_p.

The real-life acquisition process lacks some of the simplifications we used in our model. For example, samples taken in spatial and temporal proximity are correlated. To examine the performance of the sensing systems in the face of these non-ideal circumstances, we created a *discrete event simulation* [48] to compare sensed distributions to known ground truths.

As illustrated in Figure 3, we simulated a number of mobile sensors that detect nearby particles. Each sensor has a circular coverage of radius r. Collision among particles and sensors are ignored for simplicity. Sensors and particles move with uniform speeds v_{sensor} and $v_{particle}$ respectively. The simulation world is mapped as a graph, as in [49]. Each node in the graph is a traversable point by both sensors and particles and edges represent a path between the end nodes.

We assume that, in each time step, a sensor has an independent chance, p, of detecting each of the $n(x)$ persons within range along with an independent chance per location to obtain a false positive.

These assumptions lead to $\boldsymbol{N(x)}$ being sampled from the sum of a binomial distribution with mean $\boldsymbol{p}$ and a Poisson process with a given expected number $\boldsymbol{\lambda}$ A calculation of the expected value indicates that the expected value formula is satisfied and that our theoretical error calculations and bounds should be valid.

The system state can be described by various state variables: sensor and particle positions, sensor and particle waypoints, real density of particles and sensed density of particles. Sensors and particles move with a variation of the random waypoint model~\cite{johnson1996dynamic}, differing from it by the fact that sensors and particles are not allowed to change speeds; they have fixed speeds given by the system parameters$\nu_{sensor}$and $\nu_{particle}$. When a new destination is randomly picked, the trajectory on the map graph is computed using the A* algorithm~\cite{hart1968formal} and the points of the trajectory are pushed to a heap.

As time progresses, we obtain a 2D histogram for the sensed density as well as the ground truth density of particles.

Simulation Results

We evaluated different true positive rates and expected number of false positives of the sensors, p and λ and we used a Mersenne Twister pseudo number generator [47].

For the 143 possible combinations of values, we ran the simulation for 20,000 time steps 20 independent times.

The code is primarily implemented in Python with performance-sensitive sections implemented in Cython [50]. The average time to run a single experiment of this optimized code is of 11,718 seconds. A single processing and single machine processing would take roughly 1 year to run all the experiments, but running them in parallel, it took 11 days.

For each experiment we examine the decay of the metric given by the error metric equation as a function of the cumulative number of samples captured by all the sensors. We assume the error continues to decay until it reaches an asymptotic minimum error within the 20,000 simulation time steps. Afterwards, we take the average decay curve of all 20 runs for each settings configuration and take the average of the last 200 values to find the asymptotic value.

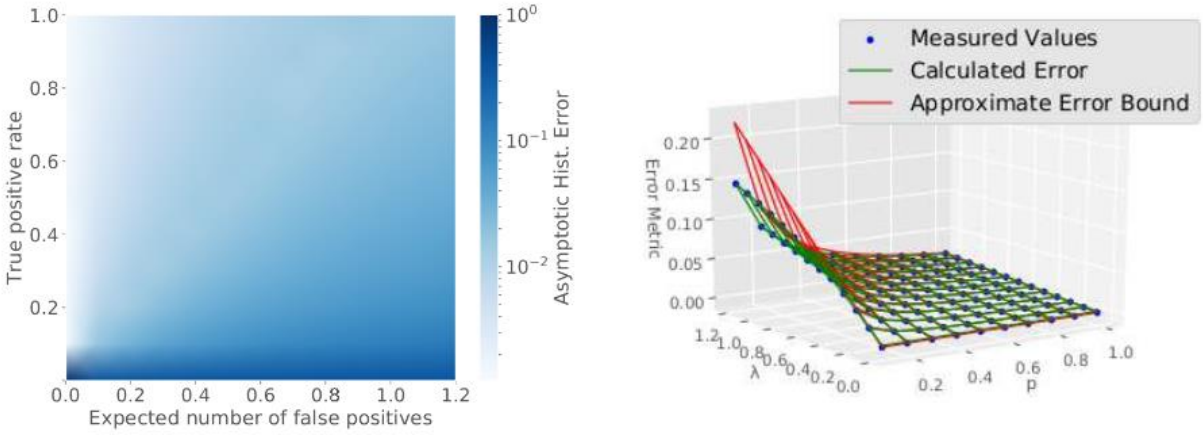

Figure 4: (Left) Asymptotic error metric between the sensed and ground truth histograms in our simulation. (Right) Comparison of the asymptotic histograms error measured from the simulation experiments to the theoretical close form and the approximate bound.

We can visualize the results from the simulations in Figure 4, which shows how the variation of true positive rate and false positive rate affect our histogram error. On the left, the asymptotic error metric between the sensed and ground truth histograms, piloted as a function of the sensors' true positive and expected number of false positives, is shown. If we take a horizontal profile of 0.2 of true positive rate, we can see how the errors are greatly affected by the variation of the expected number of false positives, varying from very low to high error values (represented by the variation on the color saturation). We compare these values to our theoretical formulas, and show they are approximately equal, as shown in Figure 4 (Right). Note that p=0 is not shown due to the denominator of the inequality equation. Finally, we show that they are within the bound given by the inequality equation.

Computer Vision Sensing

Carmera uses a fleet of camera equipped cars (such as in [51]) traveling through Manhattan to acquire a temporally and spatially dense collection of pictures. The orientation of the cameras varies and the nature of the images are similar to street level collections provided by many mapping services. However, the images are not stitched into a 360-degree panorama. Every image is accompanied by metadata including the acquisition time, location, and camera orientation. The images are captured as the vehicle travels, with no control of the content, illumination, weather, traffic conditions, or vehicular speed. The typical image depicts an urban scenario as a background and the city dynamics including pedestrians, vehicles and bicycles. Our dataset differ from several existing publications [17,18,20–22,52] by providing dense temporal coverage in addition to dense spatial coverage.

We used a sample of images captured from March 2016 to February 2017 containing 10,708,953 images. This sample presents a dense spatial sampling of the whole region over a year, but irregular

spatio-temporal sampling on a daily basis, as seen in the varying distribution of pictures by day of the week and hour of the day shown in Figure 50. All resulting heatmaps are weighted sampling according to this distribution.

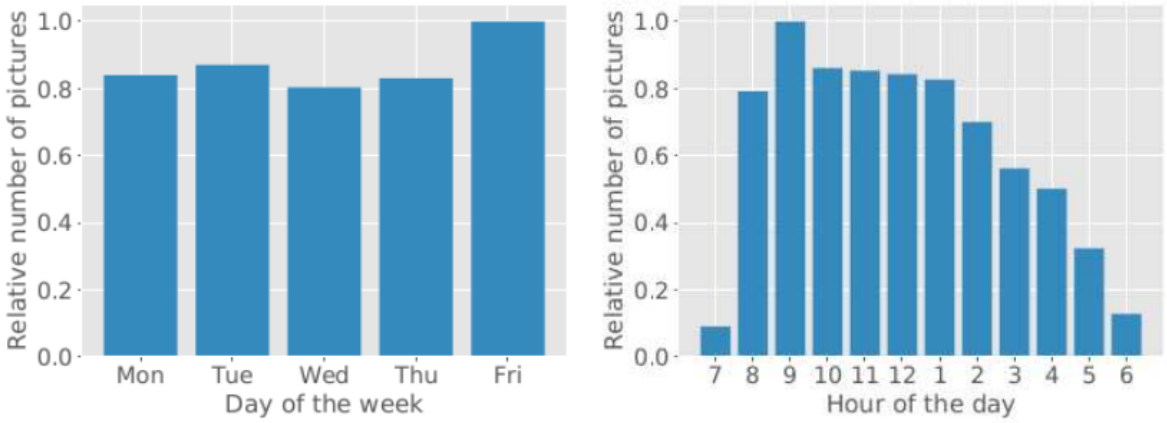

Figure 5: Distribution of pictures by day of the week and by hour of the day.

We evaluated how three computer vision algorithms for pedestrian detection perform on our dataset. The first one is based on a histogram of oriented gradients features [11]. The second one is based on the extraction of features by means of convolutional neural networks [14]. The third utilizes fully convolutional networks for accuracy and speed improvements [15].

We manually tagged 600 images to use as a ground truth. We adopt the same metric as Everingham et al. [53] when comparing the detected objects in an image to the ground-truth. A detected object is considered to correspond to a particular ground truth object if there is a minimum ratio of 50% between the overlap of the detected bounding boxes $B_{detected}$ ground-truth bounding boxes B_{gtruth}, and the union of the two areas:

$$\frac{|B_{detected} \cap B_{gtruth}|}{|B_{detected} \cup B_{gtruth}|} \geq 0.5$$

The recognition of distant objects in an image is difficult for humans and is even more difficult for computers. We assume that, on average, the size of a person within an image is an indicator of the distance from that person to the sensor and try to improve the accuracy by considering a minimal size of the people detected. Thus, bounding boxes smaller than a new hyperparameter threshold are ignored, as shown in Figure 6. When the threshold is small, as in the left image, all people in the images are annotated. As the threshold increases from medium to high in the middle and right images, the number of annotated people decreases. Those remaining tend to be closer to the camera.

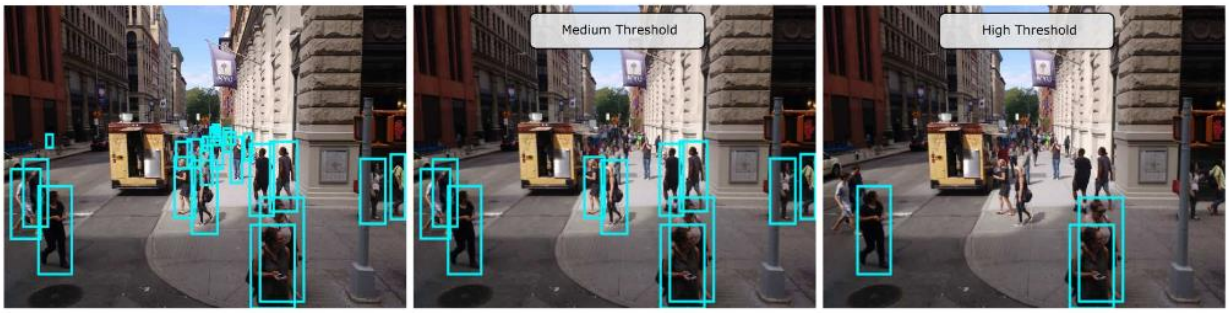

Figure 6: Variation of the ground truth annotations for different minimal person size thresholds.

As discussed below, we decided to utilize R-FCN, which we ran over our entire data set in parallel and created a database with the number of pedestrians detected in each image. This database is then aggregated in space and time to create a visualization of the pedestrian counts by finding the average number of pedestrians per image in each region.

Survey of Algorithms

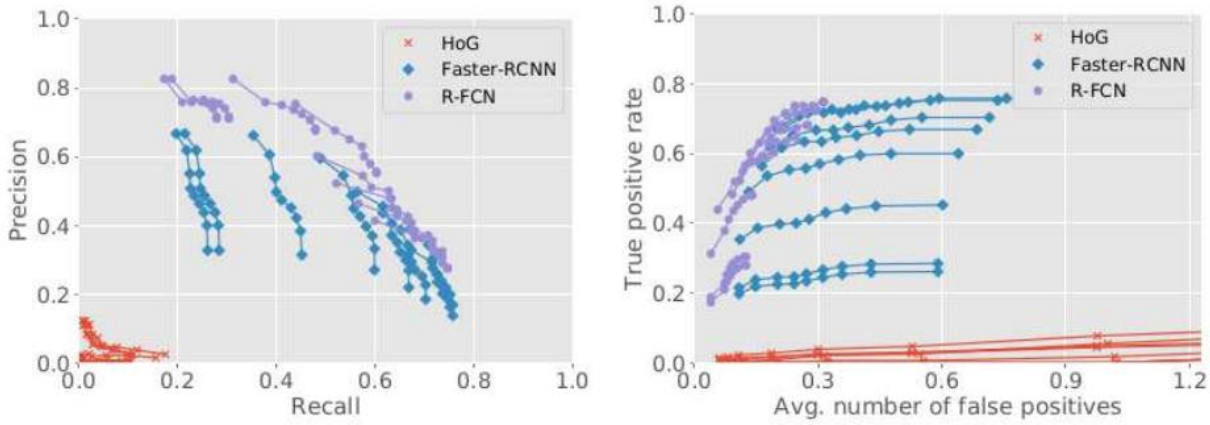

Figure 7: Comparison of HoG, Faster R-CNN and R-FCN detection on our dataset. Left: The precision and recall for each configuration of method parameters and minimum size threshold; points in the same line represents the results of the same ground-truth height threshold. In this graph the upper left corner represents an ideal algorithm. Right: The true positive rate versus the average number of false positives for the same set of parameters and ground-truth thresholds. Here, the upper left corner represents an ideal algorithm.

We used a total of 10,708,953 images, covering the region of Manhattan, Monday to Friday from 7am to 6pm. We evaluated three methods for the task of people detection [11,14,15] over a sample of our dataset. We used the Matlab [54] implementation of [11], with an 8×8 stride of the detection window, 1.05 for the pyramid scaling factor and model trained on the 96×48 resolution images from the INRIA pedestrian

dataset [11]. The detection thresholds ranged from 0.0 to 0.1, spaced by 0.01. The implementation of [14] is published by the authors and the model we used is a VGG16 network [55] trained with Pascal VOC 2007 dataset [53] with a non-maximum suppression [56] threshold of 0.3. We evaluated the method with scores ranging from 0.0 to 1.0 score, spaced by 0.1. The R-FCN algorithm [15] was also trained on the Pascal VOC 2007 dataset, but with the 101-layers neural network architecture proposed by [16]. Here again, we evaluated the method with detection scores ranging from 0.0 to 1.0, spaced by 0.1.

Figure 7 shows the results of the evaluation of the three methods over a random sample of 600 images of our dataset. The images were manually annotated and precision and recall values were computed. Ground-truth pedestrians in this comparison included tiny pedestrians, which explains such low values for recall. We can see that the overall accuracy of R-FCN was the best in our experiments. The detection times for each image are on average 5.7s for [11], 3.9s for [14] and 4.1s for [15].

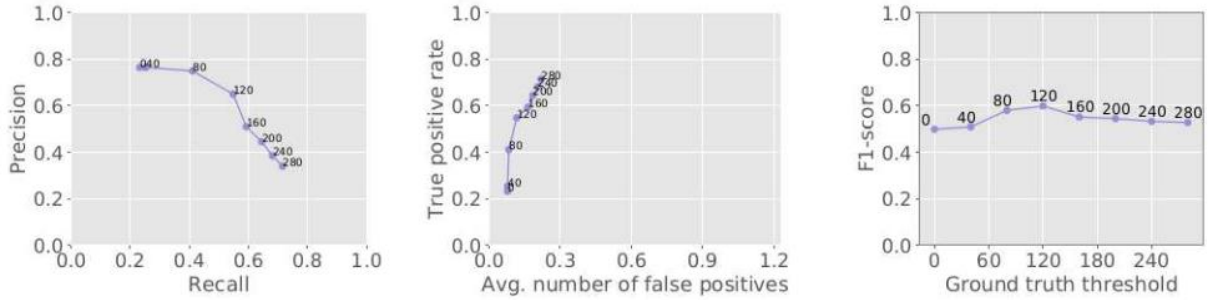

Figure 8: Evaluation of R-FCN for different ground-truth height thresholds. The utilized model has a Resnet-101 backbone trained on the Pascal VOC 2007 dataset.

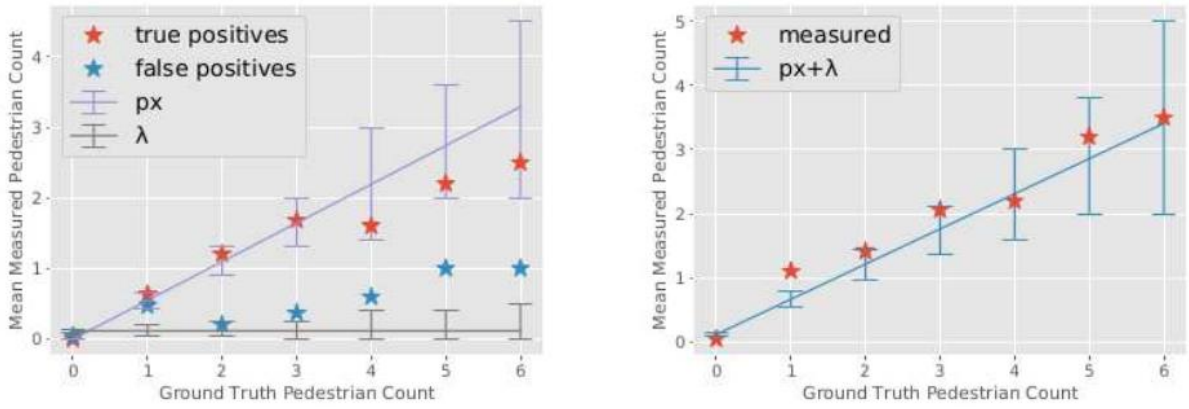

Figure 9: A comparison of the ground truth pedestrian count and the measured pedestrian count from the 600 tagged test images. While the actual true positive and false positive counts do not match their expected statistics (left), the total measured pedestrian count can be close to approximated as linear (right). It should be noted that this is only an approximation as, even taking sampling errors into

account, the mean measured count does not fit a linear model. Error bars are the 95% confidence interval of the mean.

None of the methods in Figure 7 achieve recalls exceeding 80%, and this fact is inherent to the difficulty of object detectors in detecting small objects. To mitigate this issue, the detection model we propose assumes a finite radius of coverage (see Figure 3) and thus, we establish a limit on the size of the objects detected in the image. Figure 8 shows the results of the adopted detector over our sample as we vary the minimum acceptable height. As we can see, the higher the ground-truth height threshold, the higher the precision, and specifically the recall, of the method.

Case Study

Based on the results of the previous section, we adopted a R-FCN using a residual network of 101 layers [16] trained on Pascal VOC 2007 [53], as proposed by [15]. The model was trained using a weight decay of 0.0005 and a momentum of 0.9. Assuming a method minimum score of 0.7 and height threshold of 120 pixels, 7,474,623 pedestrians were detected overall.

We compared the number of pedestrians counted as a function of the average number of ground truth pedestrians in each of the 600 manually labeled images. Error bars for the mean were computed using the 5% to 95% values of the median of the appropriate sample process. We measured the true positive rate (p) to be 0.54 and the average number of false positives (λ) to be of 0.117.

As shown in Figure 9, the actual number of true positive and false positives do not individually fit the linear and content assumptions that we proposed previously. However, the total number of pedestrians detected is closer to being approximately linear; though it still has statistically significant deviations. These stem from the vision algorithm's better-than-expected performance for images without any pedestrians and worse-than-expected performance for images with a single person. While we do not know how these deviations would affect the error bounds, we hypothesize that the two deviations would cancel themselves out and bound may still approximately hold with a slightly larger equivalent λ.

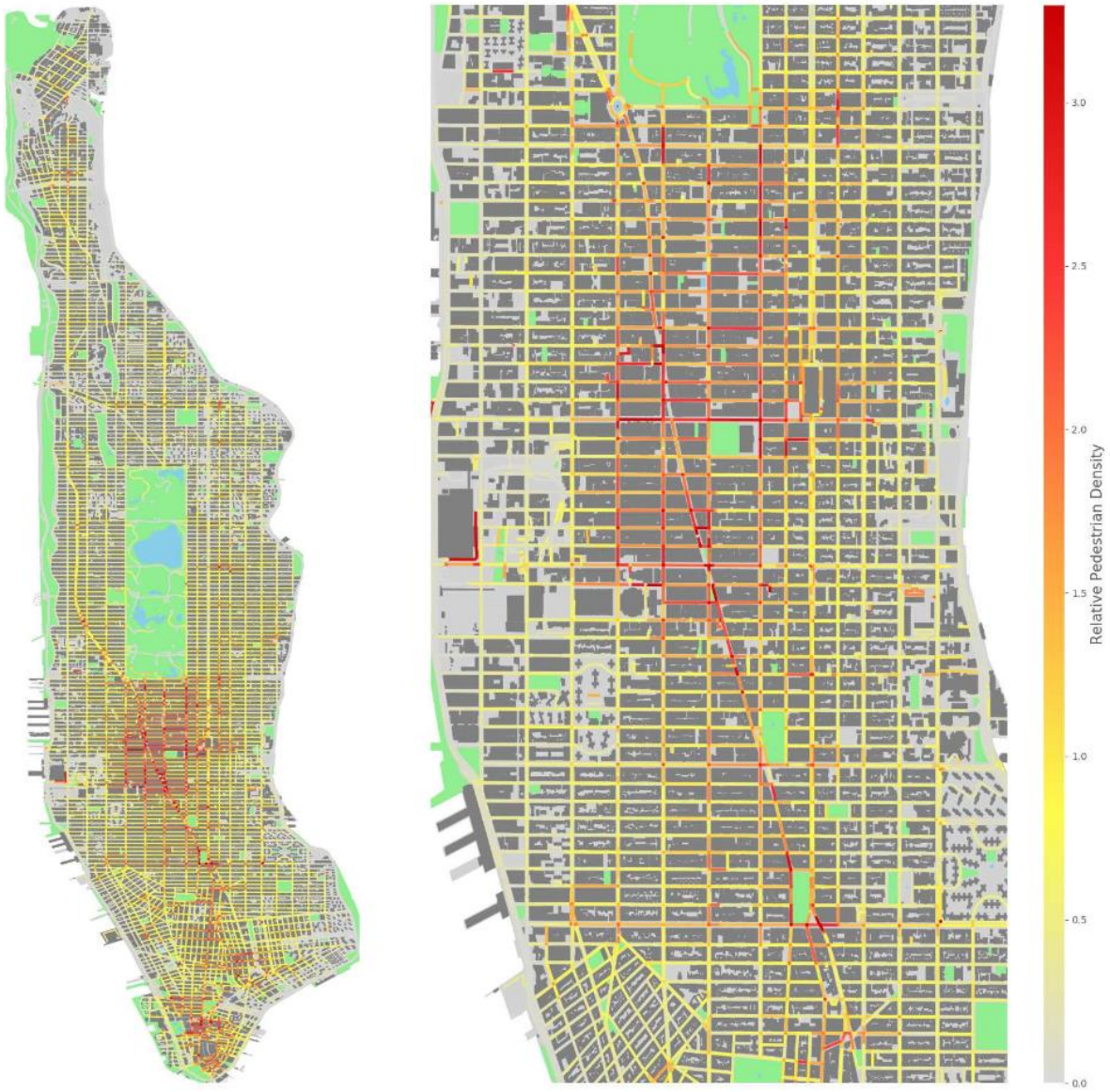

Figure 10: Visualization of the pedestrian density in Manhattan. The scale of colors represent the relative density of pedestrians. Left: The heatmap over the island of Manhattan. Right: The same heatmap enlarged to show the details of midtown and surrounding areas.

A visualization of the density of pedestrians in all of Manhattan can be seen in Figure 11, with high-density areas shown in red and lower density areas shown in yellow. For these maps, we obtained an average pedestrian density ($\hat{h}$) of 0.587, which takes us to an error of 0.062. The actual error may be larger due to the deviations from linearity discussed above.

Pedestrian distributions like ours can be useful for city planning, commercial, and other purposes. Depending on the task at hand, a high pedestrian density can be beneficial or detrimental. Taxis seeking

riders, food trucks seeking customers, and businesses seeking storefronts all benefit from large crowds. However, traffic and self-driving cars do not. Knowledge of pedestrian densities can allow city planners, civil engineers, and traffic engineers to make better decisions.

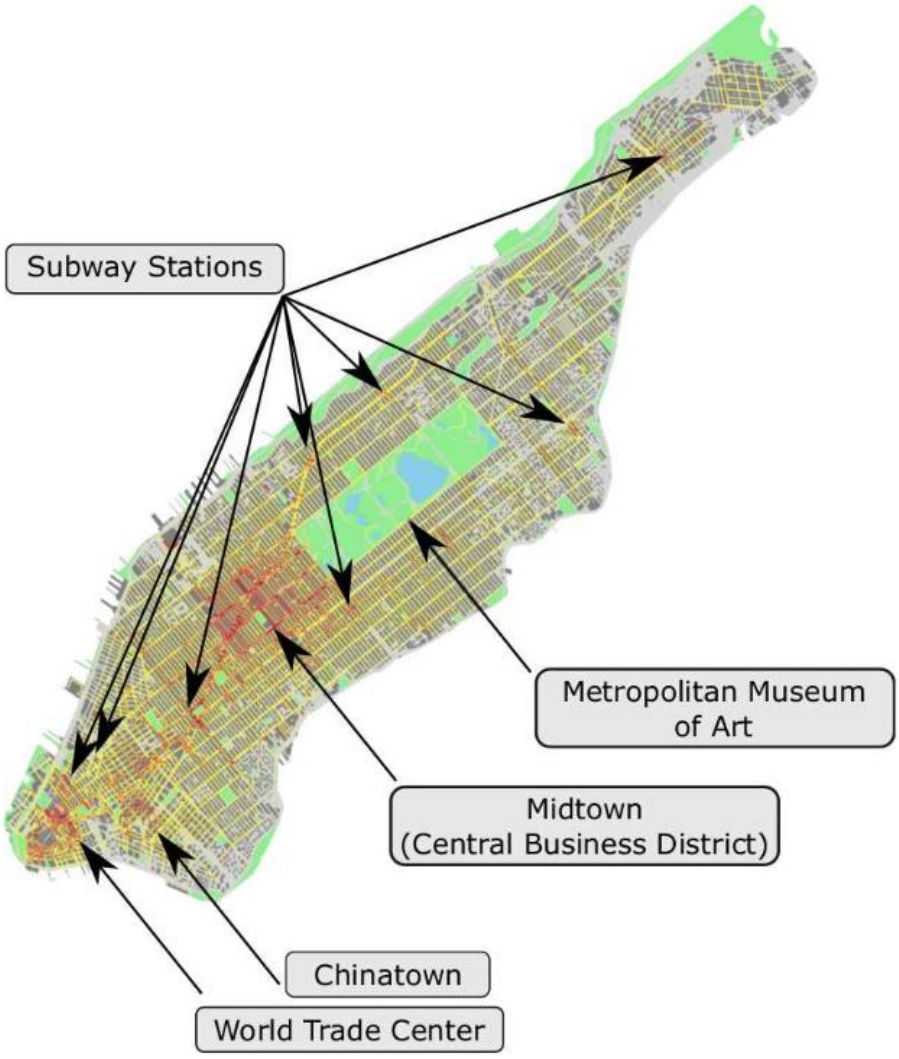

Figure 11: Map of Manhattan showing examples of locations with increased pedestrian densities. Futures studies may be able to use these correlations to better understand how cities interact with pedestrians.

Our pedestrian map can also show the effect that features of the city have on its people. As shown in Figure 12, in addition to densely populated neighborhoods, subway stations and attractions like the Metropolitan Museum of Art are all associated with a spike in the pedestrian densities. These spikes might be too localized to be detected using traditional methods. Further studies of vision-based pedestrian counts may lead to a better understanding of how cities affect people's walking habits.

Conclusion

In this project we used a large set of images from a region of Manhattan and automatically detected the number of pedestrians in each image. As a result, we obtained a map of pedestrians in the region given by the spatio-temporal sampling. Additionally, we modelled the errors in this process by simulating a sensor network with probabilistic detections. The results show that, even considering faults in the detection model, this process can still be used to get a reliable map of pedestrians in the region.

Besides the results presented, there are other potential future avenues of study, as discussed in the following section. First, we should caution that any application of our methodology should perform statistical tests to ensure that their results are statistically significant. While we set bounds on the asymptotic error after the sampling process converges, we have only provided case studies and heuristics for the time to convergence. It would be interesting to find a formal bound on time to convergence as well as provide guidelines for the appropriate statistical tests to validate the data post collection.

Our experiments could be extended to consider alternative mobility models [37], dynamics models including macroscopic ones [57–59]. We can also use data completion algorithms [36,60,61] to reconstruct a city-wide pedestrian map.

The pedestrian map generated could then be combined with other urban datasets, such as Socrata [62], weather, crime rates, census data, public transportation, bicycles and shadows [63]. We additionally aim to explore apparently disparate datasets such as wind and garbage collection.

Another future work is incorporation of advances, such as from [64] to visualize our images in the context of the city and use this visualization to gain additional insights into the other datasets analyzed in Urbane [65]. As a first pass, we are working to render the photographs in the locations they were captured. We hope to use Structure from Motion [66] to improve the accuracy of image location as well as find the orientation that the images were captured.

Additionally, we hope to use 3D popups and/or photo-based rendering to fully enhance the images in the three-dimensional environments. It is our hope that the context of the images will allow users to better understand the different datasets that are analyzed in Urbane.

Urban Portfolio

Introduction

Cities are complex systems of interrelated dynamic components tied together through a series of interactions. Transportation systems, street layouts, public utilities and land use all interact with one

another in a process that shapes and forms the city, ultimately influencing how people occupy, move and utilize the provided resources and services. Any intervention to alter one part can impact others in various ways [67]. The rapid increase in urbanization in the past century [68] has made this process much more complex, as cities struggle to satisfy the demands imposed by an ever-larger number of people with different needs and expectations. It is therefore essential to have a better understanding of different aspects of the city, and how it changes over the course of time.

The understanding and analysis of this process have been made possible in recent years with technological innovations that enable the collection of a diverse set of data that reflects how we conduct our daily lives in a city, as individuals or as a society. With an ever-growing list of urbanized areas and private firms making data publicly available through open data portals (e.g., see [69–71]); several new techniques and tools have been proposed to quantify and better understand different aspects of the cities. Most of the works are limited to understanding cities through the measurement and quantification of a subset of attributes from interesting urban data. For instance, one could use taxi trips to better understand the flow of traffic [72], sensor data to understand noise problems [73], or social media check-ins to understand land use patterns [74] or urban activity [75]. These analyses, however, are mostly limited to non-visual tabular data, and, while capturing certain aspects of the cities, fail to capture their *visual appearance*. Several attempts have been made to objectively measure features of the urban environment, many of which are perceptual or qualitative in nature and were thought of as ``unmeasurable" [76]. These attempts lead to the creation of various different audit tools [77,78]. For some time, any effort to quantify the physical appearance and built environment of cities was bound to limited geographical locations both in terms of the number of locations covered and the area of those locations, since such a task required timely and costly in-field data collection and assessment procedures [79,80].

The advent of new computer vision algorithms has made it possible to use images as a source of data to measure physical built environment characteristics [81,82]. Images provide excellent sources of information. They can encapsulate the spatial and temporal context and make otherwise abstract ideas relatable to different audiences. Using a temporally dense collection of images can make it possible to visualize not only the different blocks, neighborhoods, and boroughs of a city but also its physical changes over any period of time depending on the data availability.

Recently several private companies have been collecting street-level images with the use of cameras mounted on top of cars. Perhaps the most popular, Google Street View (GSV) [83] allows the exploration of street-level images, emphasizing the particularities of a place rather than cartographic abstractions [84]. The advent of these new GIS technologies and the availability of street view or other GIS-based images or video recordings have made it possible to conduct virtual auditing [80,85–87].This has helped expand and diversify the geographical area that can be covered in studies and reduced the amount of time and labor

required for the task to some extent. Trained experts can now directly use such image collections for analysis and data assessment.

Although virtual auditing using an image collection such as GSV has many advantages compared to in-field audits, there are still several shortcomings to this approach when it comes to the difficulty of doing the required analysis. The user still has to manually explore the image collection and identify images that are of interest for the analysis [81]. This becomes especially hard when the analysis requires identifying images satisfying conditions based on external data. For example, identifying pavements dangerous to pedestrians might involve searching only for images when there is snowfall, or analyzing the effect of construction on a neighborhood would require obtaining images in regions with a lot of construction activity restricted to the period of construction. Furthermore, the image collections provided by services such as GSV is temporally sparse, typically having images corresponding to only a single time instant. Thus, analyzing any type of temporal evolution is not possible.

Temporally Dense Street-level Images

A new data set is now available comprised of images captured in New York City over the course of a year. Unlike GSV, this collection was gathered using off-the-shelf mobile phones mounted on top of regular vehicles, without any specialized hardware, and with no guarantee that consecutive photographs were taken to uniformly cover the streets (or with the correct focus). Because there is no human intervention involved, photographs could now, however, be taken continuously. Since these vehicles are on the move most of the time throughout the year, often repeating locations, it resulted in a large number of images that are temporally dense, covering not only different times of the day but the various seasons as well. To the best of our knowledge, this is the first known data set that has comprehensive coverage of a city over both space and time.

Having such a temporally dense data set of images creates the opportunity to conduct longitudinal studies where one can identify exogenous *shocks to places* by comparing different socio-demographic indicators before and after some major natural or man-made changes. These results can then be used to answer some of the classical urban study questions [81,87]. Since this data covers different neighborhoods with similar physical features, it can enable researchers to not only test the validity of various hypotheses at a larger scale but reveal characteristics that were not obvious before.

Problem and Challenges

The goal of this work is to design a visual analysis system that allows users to effectively explore such a dense collection of image data. This system should support the following features, which were finalized based on the use case scenarios common in the workflow of an urban planning researcher.

- It should have the ability to search for images having a user desired property (which is specified through a user-provided set of query images). Note that this can be both the presence (e.g., should contain construction cones) as well as absence (should not contain construction cones) of the property. Users can use this feature to both (a) query for macro elements such as active front buildings, buildings with red-bricks facades, etc.; as well as (b) perform a city-scale search for finer details like a specific window style, or even broken curbs. Having the ability to perform such fine-grained queries opens the door to exploring a variety of research questions that are otherwise not possible.
- It should allow users to query for images based on other data sets. As mentioned earlier, this can help in policy-oriented analysis common in urban planning.
- It should allow users to query for images over regions and/or time periods of interest. This can help in a spatially and/or temporally focused analysis of the urban environment.

There are, however, several challenges in accomplishing this. First, querying for similar images having the desired properties requires first extracting the features of all the images, and then searching over this feature set. While recent advances in computer vision can be used for extracting the features, given the large number of images present, these features take hundreds of GBs of space, and thus cannot be memory resident. Second, visual exploration requires support for real-time queries, which is not possible over disk-resident data. Third, contextual queries dependent on external data sets requires interactive spatial queries over the image metadata such as the location of the image and the time it was taken. Moreover, the query interface should also support the visualization of external data sets to allow users to identify appropriate query constraints.

Motivated by the use case scenarios common in urban planning, we present a visual analysis system that enables users to contextually explore a temporally dense collection of images from a city. It allows users to visually compose contextual queries by choosing either parts of images already in the database or uploading external images. Such queries are supported by first extracting the features, as a high dimensional vector, of the images at multiple resolutions using a convolutional neural network, and then searching based on these features. To enable interactive querying using these features, we design a lightweight feature index that:

1. significantly reduces the memory footprint, thus allowing it to be memory efficient
2. converts similarity testing into simple bitwise operations, thus making it computationally efficient, requiring less than 100 ms for queries that would otherwise require tens of minutes.

Our system also supports loading and visualizing external spatiotemporal data sets, which can help to guide the user exploration.

Working in collaboration with urban planning researchers (also a co-author of this paper), we demonstrate the utility of this system through different case studies that:

- Assess pedestrian safety in NYC. In particular, check for the presence of tactile squares on curb ramps in accordance with Americans with Disabilities Act compliance, as well as pedestrian safety around construction sites.
- Explore how the visual appearance of neighborhoods has applications to urban design.
- Track the evolution of construction projects.

To the best of our knowledge, this is the first system that allows the visual exploration and analysis of a large and temporally dense collection of images from an urban center in conjunction with other urban data sets. We believe that this ability to marry a visual depiction of a city with urban data sets can significantly help and engage stakeholders when framing policies.

Related Work

We review related work in three categories: urban visual analytics, image similarity, and the use of street-level images in urban planning.

Urban Visual Analytics

The availability of an increasing number of data sets from cities has created an opportunity to analyze the city in several new ways. In order to interactively explore and analyze this data, multiple visual analytics systems have been proposed [65,72], targeted at transportation and mobility [88,89], air pollution [90], noise [73], real-estate ownership [91], and shadows [63]. A comprehensive survey on urban visual analytics is presented in Zheng et al. [3].

The use of street-level images in urban analysis has gained popularity since the introduction of GSV [83] and Microsoft's Streetside [92], which allows users to explore cities through images captured by specifically designed cameras mounted on cars. The availability of street-level images has created an opportunity to analyze cities from a new perspective. These images have been used to assess urban environments [80,93], predict street safety [82], predict urban change [94], summarize city landscapes [95], compute sky exposure [96], and detect urban features [97–100]. Arietta et al. [6] presented a method that uses street-level images to identify relationships between the visual appearance of a city and its attributes. More recently, Shen et al. proposed StreetVizor [101], using Google Street View images to analyze urban forms. Sakurada et al. [102] used a collection of images and mapping data to detect changes in buildings, applying their method to cities damaged by the tsunami in Japan. Our work differs from past proposals along three major directions. First, we use a first-of-its-kind street-level data set that is temporally dense. Second, we present a visual analysis system that enables users to explore this collection of

images by visually composing contextual queries and searching for visually similar images. Lastly, we support its spatiotemporal analysis in conjunction with other urban data sets.

Image Retrieval

The querying of large collections of images by similar features has been a major source of research in the computer vision community. Prior to 2012, image retrieval methods were mostly based on image features extracted by local descriptors such as SIFT [103] or HOG [11]. Recent neural network proposals, however, have gained attention as a viable approach for feature extraction. Two possible approaches are to either use a convolutional neural network (CNN) [55] pre-trained on image data sets, such as ImageNet, or a fine-tuned CNN model [104]. We refer the reader to the detailed survey by Zhou et al. [105] covering content-based image retrieval.

In this paper, we use an approach first presented in Razavian et al. [106]. We decided on this approach based on the accuracy it provided for our test data sets, as well as its performance, which enabled us to compute the features of an image in real time, this operation is performed when a user specifies an image as a query constraint). Their idea is to use the feature representation extracted from an intermediate layer of a pre-trained convolutional neural network.

Urban Planning

The impact of the physical appearance of the cities on various aspects of human life has been studied for years. Jane Jacobs famous “eyes on the streets” theory, explains how certain features of the built environment, such as active ground floors with lots of details and windows, townhouse stoops, greeneries or wide sidewalks can create a vibrant street life that attracts more people to public spaces and hence increases the safety [107]. Researchers from different fields have also looked at how the physical forms of an environment can shape, alter or influence certain features of daily life, such as physical and mental health [108,109], level of physical activity [110,111], willingness to walk [112], safety [110,113,114], social inclusion [115,116] and social capital [117,118].

To empirically analyze the impact of the built environment on any of the above-mentioned fields, its properties should be quantified. For this purpose, different systematic observations or audits have been designed and developed [119,120]. A majority of the physical features of urban built environments currently used in multiple audit tools were identified through direct field observations, focus groups, or experts' panels [77] or interviews [78]. These features are then added to the audit checklists or protocols, together with the objective measurements for each feature.

The most widely used method of collecting data on urban appearance is in-field auditing, which requires trained auditors to be present in the field, recording their observations based on different auditing protocols and in many cases, performing some level of on-site assessments [121–123]. However, more

recent works employed crowdsourcing and virtual auditing through GSV street-level images to investigate this relationship on a larger scale [124,125]. One of the major problems of using GSV is that the date when the images are taken is not always consistent for the whole city, or even across one neighborhood. Also, only the month and year in which each image is taken is being reported; there is no information about the specific day when the image was taken, and this can be problematic in some auditing protocols, specifically those concerned with public health or socioeconomic aspects of the built environment [80].

Our system improves upon common virtual audits by enabling the user to interactively explore a temporally dense collection of street-level images, allowing them to compose queries and search for regions with similar built environment characteristics.

Interface

The visual interface of our system consists of two main components: the *query interface* and the *exploration interface*.

Query Interface

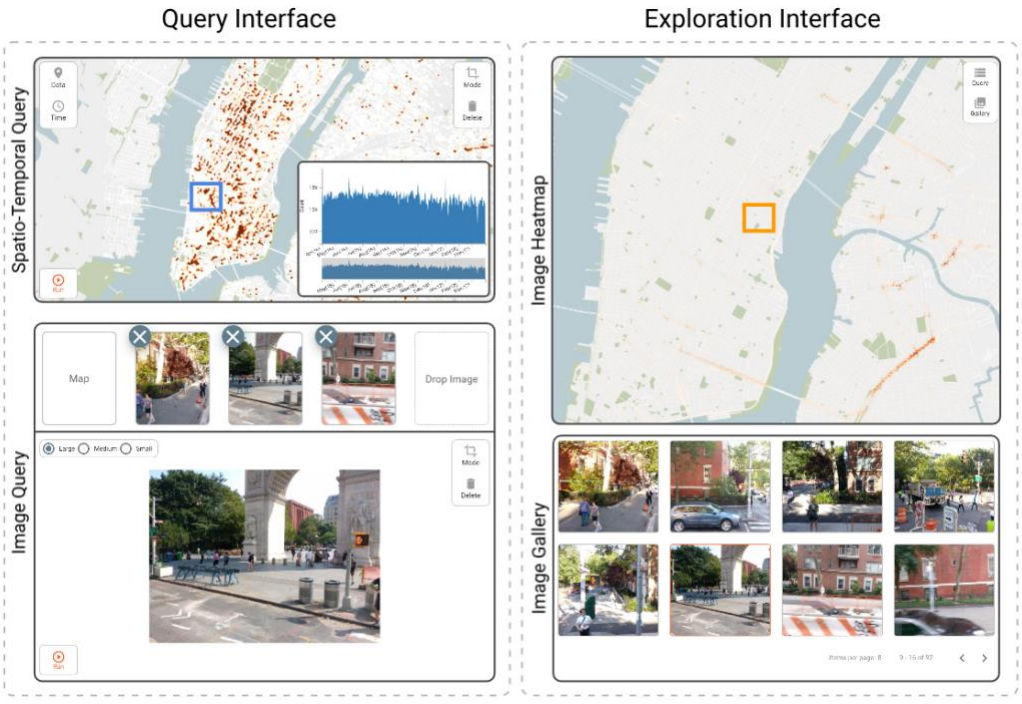

Figure 12: Query and Exploration interfaces. The Query interface allows users to compose spatial and temporal constraints based on external urban spatial and / or temporal data sets.

The query interface allows users to define two types of queries: spatiotemporal queries and image-based queries using a spatiotemporal query widget and an image query widget respectively (Figure 13). Image similarity queries can be composed by selecting images of interest, and optionally cropping regions of interest. These queries result in a set of images satisfying the different constraints. Additionally, a map also displays a heat map of the query results without the spatial constraint, thus allowing users to explore other areas having images with the required properties.

Spatio-temporal Query Widget

This widget is made of a *map view* and a *time series* view. Users can select a region of interest over the map, and a temporal range constraint using the time series.

Additionally, users can also choose a spatiotemporal data (which has been loaded into the data store) to visualize it as a heatmap as well as in the time series view. This allows users to base their spatiotemporal query constraints based on other data sets. For example, by visualizing the noise complaints data, the user can select regions having a high number of complaints and/or time periods when there is a peak in the number of complaints.

Image Query Widget

This widget allows users to compose image-based queries. A default query run without any image constraints will simply return all images satisfying the spatiotemporal constraint specified above, ordered based on time. Users can “drop” images of interest into this widget to compose image constraints. These can be either external images or from the results of a previous query. When multiple images are part of the constraint, the query returns images that satisfy all the constraints (i.e., an intersection operation is performed).

Users can also select only a patch within an image as a constraint. For example, if the user wants to query for a particular facade type, then only a small region corresponding from a larger image having this material can be cropped and used.

Depending on the user preference, they can select at what scale to perform the queries: large, medium, or small. For example, the large-scale queries can be used when searching for an urban environment of a particular type, such as open fronts or parklets, or small-scale queries when looking for small objects such as trashcans, etc. By default, we consider two image features similar if the angle between them is less than $45°$

.

Exploration Interface

The exploration interface is made up of two gallery widgets, and another map view. The first gallery view shows the list of images resulting from the query performed using the query interface. Since, the number of figures returned can be quite large and will not fit into a single view, we paginate the gallery into multiple pages. When no image constraint is provided, then the images are ordered by the time the photo was taken. Otherwise, the ranking is based on the angular distance.

Additionally, the map view in the exploration interface visualizes a heatmap of the images satisfying the image constraint over the entire data (that is, when there is not spatio-temporal constraint). This is helpful for users to quickly identify spots in the city that have properties similar to the given constraint. Users can then select a region of interest in the map and view the images from this region on the second gallery view. The use of two gallery views also helps users compare images from different regions.

Case Studies

In this section, we demonstrate the application of Urban Portfolio through four case studies set in NYC. The first case study compares different facade types common to residential areas and their distribution across NYC. The second case presents the evolution of a construction project in the neighborhood of Williamsburg. The final study assesses the presence of tactile pavings on sidewalks, as well as two common sidewalk obstacles: traffic drums and scaffolding.

Urban Design: Neighborhood Fabric

A diverse city like NYC presents a variety of different architectural styles that make up the unique urban fabric of various neighborhoods. From Brooklyn's famous brownstone row houses and Manhattan's limestone townhouses to the tall and skinny glass towers, different facades bring different rhythms to the neighborhoods. Getting a better understanding of this diverse fabric can greatly help different aspects of urban design.

For example, the design of the ground floors and facades of the buildings is proven to have a significant impact on the pedestrian's walking experience. According to Jan Gehl, fine details of the facade and display windows, narrower units and vertical facade articulation can make the walk seem shorter and less tiring. Such neighborhoods are more inviting and thus create livelier cities [126]. Detecting areas that display such characters is hence of great importance for walkability studies.

In a city like NYC, the codes and regulations for urban constructions as well as for repairs and installations differ based on the characteristics of the location where it is to be performed. For example,

safety inspections are scheduled typically every 5 years for buildings with certain facades [127]. Fixing any problems in these cases often warrants a specialized crew and/or tools. However, given the long time interval between the audits, it is difficult to identify problems early, resulting in more work and hence incurring a higher cost. This is especially true for older buildings, since they require a lot more care. Having the ability to search through updated images (indicating some of the common problems) from across the city can greatly help in performing a virtual audit, thus helping identify some problems early.

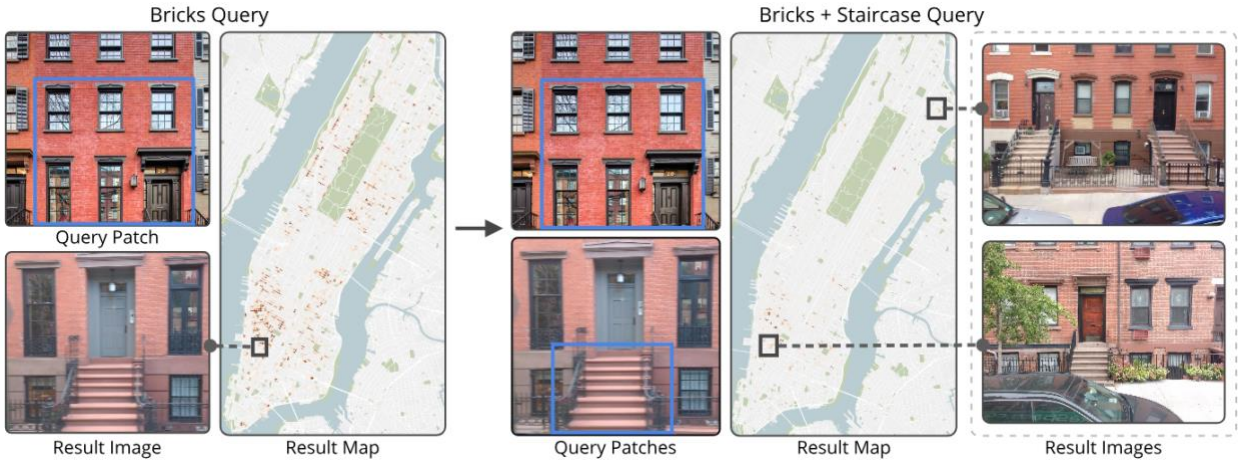

Figure 13: Exploration of similar building features across neighborhoods in Manhattan. Here, we first upload an image from a typical residential neighborhood and search for similar images using the medium level. The map on the left displays the regions of Manhattan with images that are most similar to the one used in our query. Next, we add another image to our query, now highlighting the front stoop. The map on the right shows the occurrence of similar images. We also highlight an example of two of these images.

In this case study, we show some examples of how our system can spot different architectural styles and facade materials across neighborhoods. In the first example, we use an external image of a red brick apartment with detailed windows, crop a patch of interest, and query for similar images at a medium scale (see Figure 13 left). This results in several images matching the style, as well as a heatmap visualization highlighting regions in Manhattan containing images similar to the query pattern. Note here that historical neighborhoods, such as Greenwich Village, East Village and Gramercy, present a high concentration of buildings with red bricks.

One of the result images contained a red brick apartment that has a front stoop. In the next step, we further refine our query by cropping and adding the patch corresponding to the stoop to the query constraint. Recall that using multiple images in the query constraint results in the intersection of images satisfying both the constraints. The distribution of this query result is again visualized as a heatmap. Figure 13 (right) shows representative images from two locations satisfying both the constraints. It is

interesting to see that the Greenwich village area has a higher density of such buildings compared to other areas.

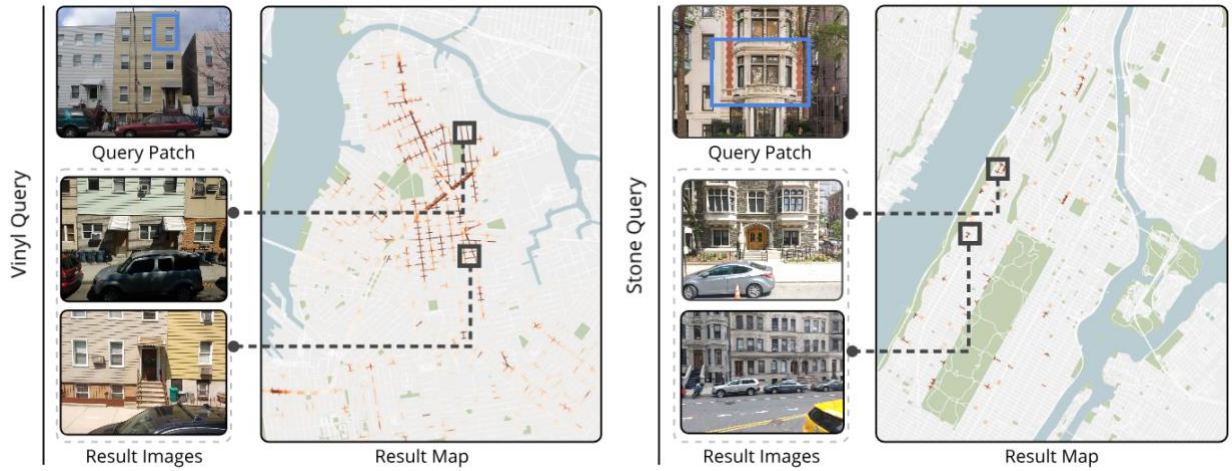

Figure 14: Querying for buildings having a (left) vinyl facade, and (right) limestone bay window of a Beaux-Arts style townhouse.

Williamsburg and Greenpoint are Brooklyn neighborhoods known for having rows of colorful vinyl townhouses. So, in the next example, shown in Figure 14 (left), we query for buildings with a vinyl facade in Brooklyn. The results show a dense cluster of vinyl townhouses in both these neighborhoods, and more sparsely spread in other locations.

In the final example, shown in Figure 14 (right), we search for a more specific detail of the building façade - the limestone bay window of a Beaux-Arts style townhouse. As seen in the resulting density distribution, this architectural feature is present in both the Upper East and West sides of Manhattan, as well as Hamilton Heights. But unlike the previous two examples, these are more sparsely and widely scattered, primarily due to the fact that these are very specific design types.

Construction Evolution

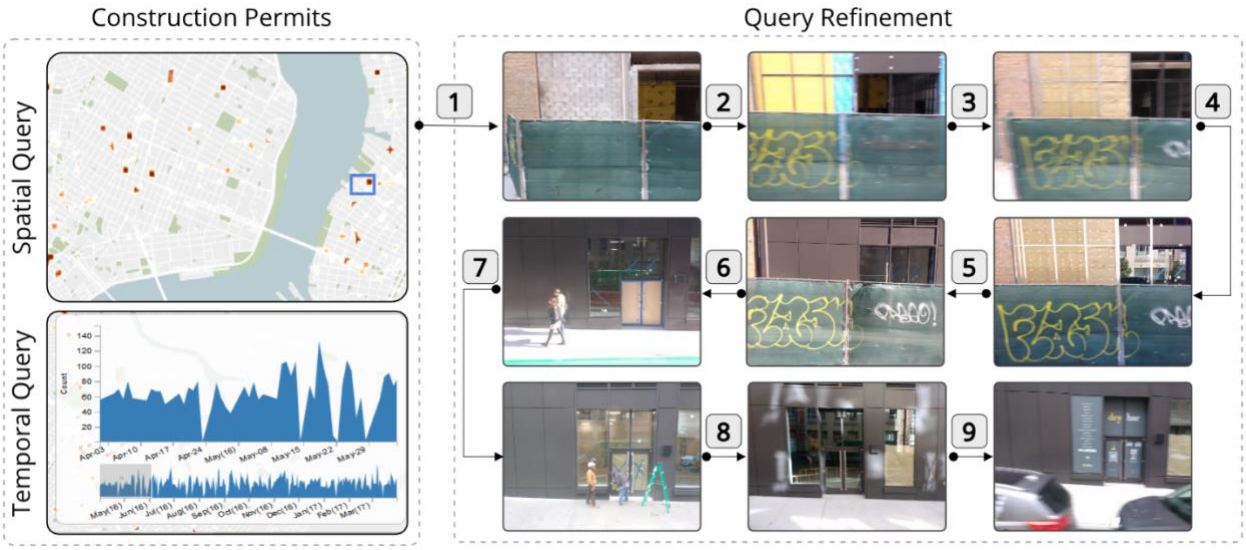

Figure 15: Example of using our tool to assess the evolution of a construction site. We start by selecting images in the beginning of 2016 from Williamsburg, Brooklyn. We use one of the images as our query, then find images later in the year when the construction is more advanced. We further use images from this stage as our query, continuously refining our query until we are able to see the inauguration of a new store in the neighborhood.

Each year, New York City's Department of Buildings (DOB) issues thousands of building permits for a variety of different purposes. DOB also provides a data set containing a list of all issued permits across all of the five boroughs of NYC. This allows for an overview of the active construction hot spots throughout the city, an increasingly important topic considering the impact of construction on noise, pedestrian safety and pollution.

However, tracking the actual evolution of construction projects and their impacts on the surrounding neighborhoods is a demanding task, considering the large number of constructions in a metropolis such as NYC. Relying only on permits to assess the evolution of a construction project can be misleading, since some permits issued by DOB may only involve minor work. Sometimes they do not even result in any construction work. In this case study, we use Urban Portfolio to demonstrate how it provides an intuitive way to visually grasp changes in a construction project over the course of a year and analyze the evolution of a construction site from the early stages of development until the end.

We started with the construction permits issued by DOB to assist in our exploration. Using the heatmap of the construction permits issued, we first filtered the images that were captured within 100 meters of an active construction site (Figure 15 (left)). We then chose the neighborhood of Williamsburg in Brooklyn for further exploration. This neighborhood has been the focus of construction initiatives in

recent years, with old houses being replaced by modern buildings. This is rapidly changing the look of the area.

After selecting a street block in Williamsburg, we query for images from the beginning of 2016. We identify an active construction site in its early stages of development and select an image to start a process of query refinement using this image. Next, we used this image to query for other such similar images in the region. One of the most similar ones is an image that was captured a few weeks in the future, depicting the construction in a more advanced stage. We then replaced the query with this new image and continued this process until we finally found an image capturing the opening of a new store (Figure 15 bottom right).

Thus, through the use of street-level images taken by an unbiased fleet of sensors, we can assess the entire evolution of the development. This way, it is possible to assess the changes in the construction site in terms of the progress of the project. It can also help with monitoring whether safety measures, like using proper pedestrian protections, are being observed, or if the sidewalks are being obstructed throughout the project. Moreover, monitoring changes in the entire block in terms of new shops, new activities or detecting early signs of gentrification in the neighborhood is also possible using our tool.

Pedestrian Safety Assessment

Americans with Disabilities Act Compliance

Figure 16: Using Urban Portfolio to inspect the presence and condition of tactile pavements in New York City, which is important to provide safe pedestrian access to people with movement or vision impairment disabilities. (a) Query image. (b) Other images in the database similar to the query image. (c) Visualizing the density of all images in NYC similar to the query image as a heatmap. The highlighted region is set as a spatial constraint to query only for other images in that region. (d) A temporal constraint is chosen based on a time period when there is rainfall. (e) An image from the result of the spatio-temporal query showing that the pavement is in dire need of repairs. Further exploration revealed that this pavement was later repaired before August 2016.

Designing inclusive, accessible streets that serve a variety of users is a crucial aspect of the human scale of the cities. Sidewalks are the most important pedestrian-dedicated planned public spaces. A proper assessment of their accessibility is then extremely important to the city and its dwellers since they are the access points to and from streets. Curbs are where the raised sidewalks or medians meet the streets, but they are often a big challenge for wheelchair users or other people with movement or vision impairment disabilities. A curb ramp, which is a short ramp cutting through a curb, can solve this problem. Title II of the Americans with Disabilities Act (ADA), requires pedestrian crossings to be accessible to people with disabilities [128]. In order to do that, specific regulations regarding the curb ramps should be met. Aside from the specifications on the size and slope of the ramp, “detectable warnings” or tactile squares must be installed on all the ramp-cuts.

Tactile squares are the distinctive yellow or red squares with bumpy surfaces designed by the Japanese inventor Seiichi Miyake in the 1960's to help visually impaired pedestrians navigate the potentially high risk and dangerous public spaces in cities more safely. The patterns are created to be detectable with canes, feet or by guide dogs. The bar patterns serve as the directional guide and the bumps show that a potentially unsafe area is close by and caution is needed.

In 2014, The Manhattan Borough President's Office conducted a study assessing ADA compatibility of around 1200 curb ramps alongside Manhattan's Broadway street. The results showed that only 115 curb cuts were fully compliant with the ADA regulations.

Following a court's approval to resolve lawsuits brought by disability rights activists, NYC has recently agreed to equip all the curbs in the city with the facilities required by the ADA and ensure that all the city's curbs will have proper ramps and tactile markings installed [129]. However, this would require an in-field assessment of the condition and quality of all of the 162,000 curbs in the city. For the 2014 study, in which less than 1 percent of the city's curbs were assessed, 40 auditors were trained and conducted the field auditing. However, given the required hours of training as well as travel costs, it becomes almost impossible to support the personnel needed not just for the initial assessment of all the curbs, but to also perform frequent audits to keep track of their conditions.

In this case study, we show how a tool such as the proposed Urban Portfolio can greatly help in performing a virtual assessment and help filter regions requiring manual intervention. We first query for images having tactile paving using an initial image with red tactile paving as our query. This enables us to visualize locations on the map (Figure 16 (c)) where tactile pavements were installed, as well as street corners where no such images were found. Figure 16 (b) shows a few example representative images from locations that had tactile pavements.

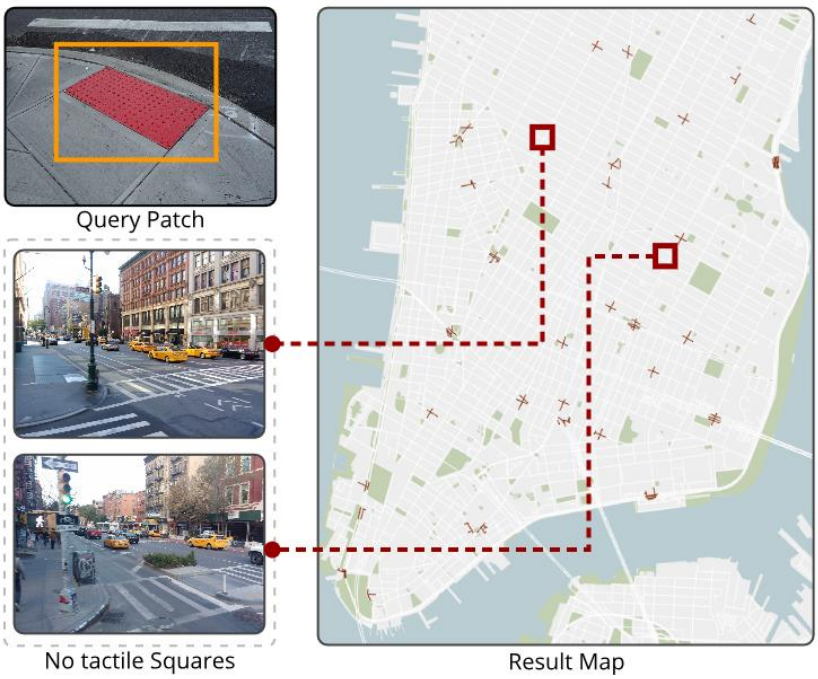

Figure 17: Exploring images from locations on the map with no tactile pavements shows that these locations indeed do not have tactile pavements.

Note that finding no images does not necessarily mean that the corresponding street corner has no tactile pavement. It could be that the location is not covered by the image database, or the image quality (e.g., the focus does not clearly capture the required feature) was not high enough to identify such a feature. So, in the next step, we restrict the spatial region to one such location and explore all images in that region. This will allow us to manually scan a small set of images to investigate why the initial query did not return any location from this region. Figure 17 shows representative images from two such locations, and we can see that in both these locations, while the curb ramps were present, the tactile pavements were indeed missing.

Similar queries can be run to assess the condition of existing tactile pavements as well. To do this, we select one of the locations having a tactile pavement and use it as a spatial constraint (Figure 16 (c)). Then, using the visualization of the weather data, we select a time range when there was rainfall (Figure 16 (d)). One of the first results resulting from this query showed an old curb with a damaged tactile pavement (see Figure 16 (e)) that has a puddle of water, which makes this curb unsafe for pedestrians. This was in a picture taken in May 2016. We then queried for images taken after the time the original query picture was taken, and it showed that the curb was then repaired, and continued to be in good condition at the end of the year.

Construction Zones Safety Compliance

Figure 18: Using Urban Portfolio to assess construction zone safety compliance. The left image shows results from a query for images containing traffic drums that are common to street renovation. The right image shows results from querying for scaffolds, commonly installed during building renovations in NYC.

Although New York City ranks high in walkability score among American cities, its sidewalks are not always easy or safe for all types of pedestrians to navigate. Oftentimes there is construction going on, and if the safety measures enacted by NYC DOB are being observed, the sidewalks are either closed or narrowed by construction cones, safety dividers or scaffolding. The width of the sidewalks in Manhattan specifically are already narrow considering its pedestrian volume. NYC's sidewalks have significantly shrunk from their size at the turn of the 20th century to accommodate more cars. Taking this into account, the city's never-ending construction projects can create a myriad of different problems for all types of pedestrians, especially those with disabilities. As much as it is important for the construction projects to comply fully with the safety measurements required by the DOB, ensuring streets' and sidewalks' accessibility for all types of users should also be a priority.

Given the amount of personnel required, regular monitoring of the safety compliance of all the construction zones and how they are impacting accessibility of adjacent streets and sidewalks is not feasible. On the other hand, having the ability to easily explore a temporally dense data set of visual images of the city can allow different stakeholders to visually assess how these safety measures are being used and what type of accessibility or safety problems they can create. This way, planning to mitigate such issues will become much easier.

In this case study, we use Urban Portfolio to query for two common sidewalk obstacles: construction cones and scaffolding. We first query for all images near a construction site using the heatmap of the construction permits data. We then selected an image that has one of these obstacles; we then crop this image around the object and query for similar objects, querying for small scale similarity for

construction cones (see Figure 18 left), and medium scale similarity for scaffolding (see Figure 18 right). This query scale was chosen based on the size of the object (patch) we were querying.

The map in the exploration view denotes the locations in Manhattan where there is an active construction site with these two objects, allowing stakeholders to visually inspect these sites and check whether these structures conform to the regulated norms without the need for manual intervention. Figure 18 shows representative images from three such sites containing a construction cone, and three having scaffolding.

Conclusion

Glass facades

A particular type of query that failed often was a query for glass facades. This was primarily because such buildings reflected the local environment, and thus do not have a unique set of features like opaque materials. As a next step, we would like to explore strategies that can help in identifying if a particular building has a glass facade or not. This could potentially involve trying to identify features that are clearly reflections.

Quality of similarity search query results

The quality of similarity query results is impacted by many factors such as lighting and distance from the object. But one particular property which affected the results the most was the angle at which the picture was taken. Overcoming this would involve searching over different rotations of the picture content, which is not an easy operation. Another factor affecting the similarity results is the similarity threshold. We used a threshold of 45 degrees by default and found it to be sufficient for most of the cases we tested. It would, however, be interesting to explore ways to automatically estimate this parameter based on the quality of the query images.

Conclusions

The availability of spatially and temporally dense street-level images opens new vistas for urban planning, design, and policy by making it easy to virtually audit different characteristics of an urban environment. However, the large size of such data sets creates several challenges to enable interactive analysis. In this work we presented Urban Portfolio, a visual analysis tool that allows for interactive searching and exploration of such data sets. Context-based searching is accomplished using features extracted from a convolutional neural network. A novel index that makes use of bit-wise operations to compute similarity between the features was proposed, which not only reduced the memory overheard by over 2 orders of magnitude, but also improved the query times by over 5 orders of magnitude, thus allowing for interactive queries crucial for visualization systems.

Working in collaboration with an urban planning researcher, we also demonstrated the potential utility of such a system through several case studies. We believe that this tool can have a significant impact on the way cities go about monitoring and assessing the urban environment by not only reducing the cost of these operations but also greatly simplifying this process.

References

1. Raasveldt M, Mühleisen H. MonetDBLite: An Embedded Analytical Database. *CoRR*. 2018;abs/1805.08520. http://arxiv.org/abs/1805.08520.

2. Zacharatou ET, Doraiswamy H, Ailamaki A, Silva CT, Freiref J. GPU rasterization for real-time spatial aggregation over arbitrary polygons. *Proceedings of the VLDB Endowment*. 2017;11(3):352–365.

3. Zheng Y, Capra L, Wolfson O, Yang H. Urban computing: concepts, methodologies, and applications. *ACM Transactions on Intelligent Systems and Technology (TIST)*. 2014;5(3):38.

4. Data –USEEPAAD USEPAA. *(Https://Www3.Epa.Gov/Airdata/Ad_data_daily.Html)*. United States Environment Protection Agency; 2017.

5. Vanegas CA, Aliaga DG, Benes B. Automatic extraction of Manhattan-world building masses from 3D laser range scans. *Visualization and Computer Graphics, IEEE Transactions on*. 2012;18(10):1627–1637.

6. Arietta SM, Efros AA, Ramamoorthi R, Agrawala M. City forensics: Using visual elements to predict non-visual city attributes. *IEEE Transactions on Visualization and Computer Graphics*. 2014;20(12):2624–2633.

7. Google Inc. *(Https://Maps.Google.Com)*. Google Maps; 2017.

8. Burges CJ. A tutorial on support vector machines for pattern recognition. *Data mining and knowledge discovery*. 1998;2(2):121–167.

9. Whyte WH. *City: Rediscovering the Center*. Pennsylvania, USA: University of Pennsylvania Press; 2012.

10. Reades J, Calabrese F, Sevtsuk A, Ratti C. Cellular census: Explorations in urban data collection. *IEEE Pervasive computing*. 2007;6(3):30–38.

11. Dalal N, Triggs B. Histograms of oriented gradients for human detection. In: *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. Vol 1. ; 2005:886-893 vol. 1.

12. Krizhevsky A, Sutskever I, Hinton GE. Imagenet classification with deep convolutional neural networks. In: *Advances in Neural Information Processing Systems*. Nevada, USA; 2012:1097–1105.

13. Szegedy C, Liu W, Jia Y, et al. Going deeper with convolutions. In: *Computer Vision and Pattern Recognition, 2015. CVPR 2015. IEEE Conference On*. Massachusetts, USA; 2015.

14. Ren S, He K, Girshick R, Sun J. Faster R-CNN: Towards real-time object detection with region proposal networks. In: *Advances in Neural Information Processing Systems*. ; 2015:91–99.

15. Dai J, Li Y, He K, Sun J. R-FCN: Object Detection via Region-based Fully Convolutional Networks. *arXiv preprint arXiv:160506409*. 2016.

16. He K, Zhang X, Ren S, Sun J. Deep Residual Learning for Image Recognition. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition*. ; 2016:770-778.

17. Oh S, Hoogs A, Perera A, et al. A large-scale benchmark dataset for event recognition in surveillance video. In: *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference On*. Colorado, USA: IEEE; 2011:3153–3160.

18. Tokuda EK, Ferreira GBA, Silva C, Cesar-Jr RM. A novel semi-supervised detection approach with weak annotation. In: *Image Analysis and Interpretation (SSIAI), 2018 IEEE Southwest Symposium On*. Nevada, USA: IEEE; 2018.

19. Vezzani R, Cucchiara R. ViSOR: Video surveillance on-line repository for annotation retrieval. In: *Multimedia and Expo, 2008 IEEE International Conference On*. IEEE; 2008:1281–1284.

20. Cordts M, Omran M, Ramos S, et al. The cityscapes dataset for semantic urban scene understanding. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. Nevada, USA: IEEE; 2016:3213–3223.

21. Geiger A, Lenz P, Stiller C, Urtasun R. Vision meets Robotics: The KITTI Dataset. *International Journal of Robotics Research (IJRR)*. 2013.

22. Maddern W, Pascoe G, Linegar C, Newman P. 1 year, 1000 km: The Oxford RobotCar dataset. *The International Journal of Robotics Research*. 2017;36(1):3–15.

23. Akyildiz IF, Melodia T, Chowdhury KR. A survey on wireless multimedia sensor networks. *Computer networks*. 2007;51(4):921–960.

24. Akyildiz IF, Su W, Sankarasubramaniam Y, Cayirci E. A survey on sensor networks. *IEEE Communications magazine*. 2002;40(8):102–114.

25. Othman MF, Shazali K. Wireless sensor network applications: A study in environment monitoring system. *Procedia Engineering*. 2012;41:1204–1210.

26. Wang G, Cao G, LaPorta T. A bidding protocol for deploying mobile sensors. In: *Network Protocols, 2003. Proceedings. 11th IEEE International Conference On*. Georgia, USA: IEEE; 2003:315–324.

27. Huang C-F, Tseng Y-C. The coverage problem in a wireless sensor network. *Mobile Networks and Applications*. 2005;10(4):519–528.

28. Yang DB, Guibas LJ, others. Counting people in crowds with a real-time network of simple image sensors. In: *Proceedings of the IEEE International Conference on Computer Vision Workshops*. Nice, France: IEEE; 2003:122.

29. Basagni S, Carosi A, Petrioli C. Controlled vs. uncontrolled mobility in wireless sensor networks: Some performance insights. In: *Vehicular Technology Conference, 2007. VTC-2007 Fall. 2007 IEEE 66th*. Maryland, USA: IEEE; 2007:269–273.

30. Karagiorgou S, Pfoser D, Skoutas D. A layered approach for more robust generation of road network maps from vehicle tracking data. *ACM Transactions on Spatial Algorithms and Systems (TSAS)*. 2017;3(1):3.

31. Shi W, Shen S, Liu Y. Automatic generation of road network map from massive GPS, vehicle trajectories. In: *Intelligent Transportation Systems, 2009. ITSC'09. 12th International IEEE Conference On*. Missouri, USA: IEEE; 2009:1–6.

32. Lane ND, Miluzzo E, Lu H, Peebles D, Choudhury T, Campbell AT. A survey of mobile phone sensing. *IEEE Communications magazine*. 2010;48(9):140–150.

33. Rana RK, Chou CT, Kanhere SS, Bulusu N, Hu W. Ear-phone: an end-to-end participatory urban noise mapping system. In: *Proceedings of the 9th ACM/IEEE International Conference on Information Processing in Sensor Networks*. ACM; 2010:105–116.

34. Sheng X, Tang J, Zhang W. Energy-efficient collaborative sensing with mobile phones. In: *INFOCOM, 2012 Proceedings IEEE*. Florida, USA: IEEE; 2012:1916–1924.

35. Consolvo S, McDonald DW, Toscos T, et al. Activity sensing in the wild: a field trial of ubifit garden. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. Florence,Italy: ACM; 2008:1797–1806.

36. Li W, Wolinski D, Lin MC. City-scale traffic animation using statistical learning and metamodel-based optimization. *ACM Transactions on Graphics (TOG)*. 2017;36(6):200.

37. Camp T, Boleng J, Davies V. A survey of mobility models for ad hoc network research. *Wireless communications and mobile computing*. 2002;2(5):483–502.

38. Davies VA, others. Evaluating mobility models within an ad hoc network. 2000.

39. Johnson DB, Maltz DA. Dynamic source routing in ad hoc wireless networks. *Mobile computing*. 1996;353(1):153–181.

40. Liang B, Haas ZJ. Predictive distance-based mobility management for PCS networks. In: *INFOCOM'99. Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*. Vol 3. New York, USA: IEEE; 1999:1377–1384.

41. Lesser V, Ortiz Jr CL, Tambe M. *Distributed Sensor Networks: A Multiagent Perspective*. Vol 9. Springer Science & Business Media; 2012.

42. Niazi MA, Hussain A. A novel agent-based simulation framework for sensing in complex adaptive environments. *IEEE Sensors Journal*. 2011;11(2):404–412.

43. Vinyals M, Rodriguez-Aguilar JA, Cerquides J. A survey on sensor networks from a multiagent perspective. *The Computer Journal*. 2011;54(3):455–470.

44. Titzer BL, Lee DK, Palsberg J. Avrora: Scalable sensor network simulation with precise timing. In: *Information Processing in Sensor Networks, 2005. IPSN 2005. Fourth International Symposium On*. Tennessee, USA: IEEE; 2005:477–482.

45. Knuth DE. *The Art of Computer Programming*. Vol 3. Pearson Education; 1997.

46. Park SK, Miller KW. Random number generators: good ones are hard to find. *Communications of the ACM*. 1988;31(10):1192–1201.

47. Matsumoto M, Nishimura T. Mersenne twister: a 623-dimensionally equidistributed uniform pseudo-random number generator. *ACM Transactions on Modeling and Computer Simulation (TOMACS)*. 1998;8(1):3–30.

48. Law AM, Kelton WD, Kelton WD. *Simulation Modeling and Analysis*. Vol 3. Arizona, USA: McGraw-Hill New York; 2007.

49. Tian J, Hahner J, Becker C, Stepanov I, Rothermel K. Graph-based mobility model for mobile ad hoc network simulation. In: *Simulation Symposium, 2002. Proceedings. 35th Annual*. California, USA: IEEE; 2002:337–344.

50. Behnel S, Bradshaw R, Citro C, Dalcin L, Seljebotn DS, Smith K. Cython: The Best of Both Worlds. *Computing in Science Engineering*. 2011;13(2):31-39. doi:10.1109/MCSE.2010.118

51. Lee U, Magistretti E, Gerla M, Bellavista P, Corradi A. Dissemination and harvesting of urban data using vehicular sensing platforms. *IEEE Transactions on Vehicular Technology*. 2009;58(2):882–901.

52. Vezzani R, Cucchiara R. Video surveillance online repository (visor): an integrated framework. *Multimedia Tools and Applications*. 2010;50(2):359–380.

53. Everingham M, Van Gool L, Williams CK, Winn J, Zisserman A. The pascal visual object classes (voc) challenge. *International journal of computer vision*. 2010;88(2):303–338.

54. The MathWorks, Inc. *Matlab Version 2017b*. Massachusetts, USA; 2017.

55. Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:14091556*. 2014.

56. Kitchen L, Rosenfeld A. Gray-level corner detection. *Pattern recognition letters*. 1982;1(2):95–102.

57. Helbing D. A fluid dynamic model for the movement of pedestrians. *arXiv preprint cond-mat/9805213*. 1998.

58. Helbing D. Traffic and related self-driven many-particle systems. *Reviews of modern physics*. 2001;73(4):1067.

59. Iwata T, Shimizu H, Naya F, Ueda N. Estimating People Flow from Spatiotemporal Population Data via Collective Graphical Mixture Models. *ACM Transactions on Spatial Algorithms and Systems (TSAS)*. 2017;3(1):2.

60. Gandy S, Recht B, Yamada I. Tensor completion and low-n-rank tensor recovery via convex optimization. *Inverse Problems*. 2011;27(2):025010.

61. Li L, Li Y, Li Z. Efficient missing data imputing for traffic flow by considering temporal and spatial dependence. *Transportation research part C: emerging technologies*. 2013;34:108–120.

62. NYC open data. *(Https://Opendata.Cityofnewyork.Us/)*. New York city open data portal; 2017.

63. Miranda F, Doraiswamy H, Lage M, Wilson L, Hsieh M, Silva CT. Shadow Accrual Maps: Efficient Accumulation of City-Scale Shadows Over Time. *IEEE Transactions on Visualization and Computer Graphics*. 2019;25(3):1559-1574.

64. Photosynth. *(Https://Blogs.Msdn.Microsoft.Com/Photosynth/ 2017/02/06/Microsoft-Photosynth-Has-Been-Shut-Down/)*. Microsoft Photosynth; 2017.

65. Doraiswamy H, Tzirita Zacharatou E, Miranda F, et al. Interactive Visual Exploration of Spatio-Temporal Urban Data Sets Using Urbane. In: *Proceedings of the 2018 International Conference on Management of Data*. ; 2018:1693–1696. http://doi.acm.org/10.1145/3183713.3193559.

66. Koenderink JJ, Van Doorn AJ. Affine structure from motion. *JOSA A*. 1991;8(2):377–385.

67. Batty M. Cities as Complex Systems: Scaling, Interaction, Networks, Dynamics and Urban Morphologies. 2009.

68. UN Department of Economic and Social Affairs. *UN 2018 Revision of World Urbanization Prospects*.

69. Barbosa L, Pham K, Silva C, Vieira MR, Freire J. Structured open urban data: understanding the landscape. *Big data*. 2014;2(3):144–154.

70. *Twitter Public API*. https://dev.twitter.com/streaming.

71. *Yahoo Labs*.; 2018. https://webscope.sandbox.yahoo.com/.

72. Ferreira N, Poco J, Vo HT, Freire J, Silva CT. Visual exploration of big spatio-temporal urban data: A study of New York City taxi trips. *IEEE Transactions on Visualization and Computer Graphics*. 2013;19(12):2149–2158.

73. Miranda F, Lage M, Doraiswamy H, et al. Time Lattice: A Data Structure for the Interactive Visual Analysis of Large Time Series. *Computer Graphics Forum*. 2018;37(3):23-35. doi:10.1111/cgf.13398

74. Quercia D, Saez D. Mining urban deprivation from foursquare: Implicit crowdsourcing of city land use. *IEEE Pervasive Computing*. 2014;13(2):30–36.

75. Miranda F, Doraiswamy H, Lage M, et al. Urban Pulse: Capturing the Rhythm of Cities. *IEEE Transactions on Visualization and Computer Graphics*. 2017;23(1):791-800. doi:10.1109/TVCG.2016.2598585

76. Ewing R, Handy S. Measuring the Unmeasurable: Urban Design Qualities Related to Walkability. *Journal of Urban Design*. 2009;14(1):65-84.

77. Day K, Boarnet M, Alfonzo M, Forsyth A. The Irvine–Minnesota Inventory to Measure Built Environments: Development. *American Journal of Preventive Medicine*. 2006;30(2):144-152.

78. Pikora T, Giles-Corti B, Bull F, Jamrozik K, Donovan R. Developing a framework for assessment of the environmental determinants of walking and cycling. *Social science & medicine*. 2003;56(8):1693–1703.

79. Purciel M, Neckerman KM, Lovasi GS, et al. Creating and validating GIS measures of urban design for health research. *Journal of Environmental Psychology*. 2009;29(4):457-466.

80. Rundle AG, Bader MDM, Richards CA, Neckerman KM, Teitler JO. Using Google Street View to audit neighborhood environments. *American Journal of Preventive Medicine*. 2011;40(1):94–100.

81. Glaeser EL, Kominers SD, Luca M, Naik N. Big data and big cities: The promises and limitations of improved measures of urban life. *Economic Inquiry*. 2018;56(1):114–137.

82. Naik N, Philipoom J, Raskar R, Hidalgo C. Streetscore-predicting the perceived safety of one million streetscapes. In: *Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition Workshops*. ; 2014:779–785.

83. Anguelov D, Dulong C, Filip D, et al. Google Street View: Capturing the world at street level. *Computer*. 2010;43(6):32–38.

84. Shapiro A. Street-level: Google Street View's abstraction by datafication. *New Media & Society*. 2017.

85. Badland HM, Opit S, Witten K, Kearns RA, Mavoa S. Can virtual streetscape audits reliably replace physical streetscape audits? *Journal of Urban Health*. 2010;87(6):1007–1016.

86. Badland HM, Schofield GM, Witten K, et al. Understanding the Relationship between Activity and Neighbourhoods (URBAN) Study: research design and methodology. *BMC Public Health*. 2009;9(1):224.

87. Kelly CM, Wilson JS, Baker EA, Miller DK, Schootman M. Using Google Street View to audit the built environment: inter-rater reliability results. *Annals of Behavioral Medicine*. 2012;45(suppl_1):S108–S112.

88. Andrienko G, Andrienko N. Spatio-temporal aggregation for visual analysis of movements. In: *2008 IEEE Symposium on Visual Analytics Science and Technology*. ; 2008:51-58.

89. Zeng W, Fu C, Arisona SM, Erath A, Qu H. Visualizing Mobility of Public Transportation System. *IEEE Transactions on Visualization and Computer Graphics*. 2014;20(12):1833-1842.

90. Qu H, Qu H, Qu H, et al. Visual Analysis of the Air Pollution Problem in Hong Kong. *IEEE Transactions on Visualization and Computer Graphics*. 2007;13(6):1408-1415.

91. Hoang-Vu T-A, Been V, Ellen IG, Weselcouch M, Freire J. Towards Understanding Real-Estate Ownership in New York City: Opportunities and Challenges. In: *Proceedings of the International Workshop on Data Science for Macro-Modeling*. DSMM'14. New York, NY, USA: ACM; 2014:15:1–15:2. http://doi.acm.org/10.1145/2630729.2630746.

92. *StreetSide: Dynamic Street-Level Imagery via Bing Maps*.

93. Li X, Zhang C, Li W, Ricard R, Meng Q, Zhang W. Assessing street-level urban greenery using Google Street View and a modified green view index. *Urban Forestry & Urban Greening*. 2015;14(3):675–685.

94. Naik N, Kominers SD, Raskar R, Glaeser EL, Hidalgo CA. Computer vision uncovers predictors of physical urban change. *Proceedings of the National Academy of Sciences*. 2017;114(29):7571–7576.

95. Doersch C, Singh S, Gupta A, Sivic J, Efros AA. What Makes Paris Look like Paris? *ACM Transactions on Graphics*. 2012;31(4):101:1–101:9.

96. Carrasco-Hernandez R, Smedley ARD, Webb AR. Using urban canyon geometries obtained from Google Street View for atmospheric studies: Potential applications in the calculation of street level total shortwave irradiances. *Energy and Buildings*. 2015;86:340–348.

97. Balali V, Rad AA, Golparvar-Fard M. Detection, classification, and mapping of US traffic signs using google street view images for roadway inventory management. *Visualization in Engineering*. 2015;3(1):15.

98. Hara K, Sun J, Moore R, Jacobs D, Froehlich J. Tohme: detecting curb ramps in Google Street View using crowdsourcing, computer vision, and machine learning. In: *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology*. ACM; 2014:189–204.

99. Lander C, Wiehr F, Herbig N, Krüger A, Löchtefeld M. Inferring landmarks for pedestrian navigation from mobile eye-tracking data and Google Street View. In: *Proceedings of the 2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems*. ; 2017:2721–2729.

100. Zhang Y, Dong R. Impacts of Street-Visible Greenery on Housing Prices: Evidence from a Hedonic Price Model and a Massive Street View Image Dataset in Beijing. *ISPRS International Journal of Geo-Information*. 2018;7(3).

101. Shen Q, Zeng W, Ye Y, et al. StreetVizor: Visual Exploration of Human-Scale Urban Forms Based on Street Views. *IEEE Transactions on Visualization and Computer Graphics*. 2018;24(1):1004–1013.

102. Sakurada K, Tetsuka D, Okatani T. Temporal city modeling using street level imagery. *Computer Vision and Image Understanding*. 2017;157:55–71.

103. Lowe DG. Distinctive Image Features from Scale-Invariant Keypoints. *Int J Comput Vision*. 2004;60(2):91–110.

104. Babenko A, Slesarev A, Chigorin A, Lempitsky V. Neural codes for image retrieval. In: *European Conference on Computer Vision*. Springer; 2014:584–599.

105. Zhou W, Li H, Tian Q. Recent Advance in Content-based Image Retrieval: A Literature Survey. *arXiv preprint arXiv:170606064*. 2017.

106. Sharif Razavian A, Azizpour H, Sullivan J, Carlsson S. CNN features off-the-shelf: an astounding baseline for recognition. In: *Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition Workshops*. ; 2014:806–813.

107. Jacobs J. The death and life of great American cities. 1961. *New York: Vintage*. 1992.

108. Lee BJ, Jang TY, Wang W, Namgung M. Design criteria for an urban sidewalk landscape considering emotional perception. *Journal of urban planning and development*. 2009;135(4):133–140.

109. Nickelson J, Wang AR, Mitchell QP, Hendricks K, Paschal A. Inventory of the physical environment domains and subdomains measured by neighborhood audit tools: A systematic literature review. *Journal of Environmental Psychology*. 2013;36:179-189.

110. Forsyth A, Hearst M, Oakes JM, Schmitz KH. Design and Destinations: Factors Influencing Walking and Total Physical Activity. *Urban Studies*. 2008;45(9):1973-1996.

111. Williams JE, Evans M, Kirtland KA, et al. Development and Use of a Tool for Assessing Sidewalk Maintenance as an Environmental Support of Physical Activity. *Health Promotion Practice*. 2005;6(1):81-88.

112. Katzmarzyk PT, Denstel KD, Beals K, et al. Results from the United States 2018 Report Card on Physical Activity for Children and Youth. *Journal of Physical Activity and Health*. 2018;15(S2):S422-S424.

113. Aghaabbasi M, Moeinaddini M, Shah MZ, Asadi-Shekari Z, Kermani MA. Evaluating the capability of walkability audit tools for assessing sidewalks. *Sustainable Cities and Society*. 2018;37:475-484.

114. Asadi-Shekari Z, Moeinaddini M, Shah MZ. Pedestrian safety index for evaluating street facilities in urban areas. *Safety Science*. 2015;74:1-14.

115. Bise RD, Rodgers III JC, Maguigan MA, et al. Sidewalks as Measures of Infrastructure Inequities. *Southeastern Geographer*. 2018;58(1):39–57.

116. Thornton CM, Conway TL, Cain KL, et al. Disparities in pedestrian streetscape environments by income and race/ethnicity. *SSM-population health*. 2016;2:206–216.

117. Rogers SH, Gardner KH, Carlson CH. Social capital and walkability as social aspects of sustainability. *Sustainability*. 2013;5(8):3473–3483.

118. Toker Z. Walking beyond the Socioeconomic Status in an objectively and perceptually walkable pedestrian environment. *Urban studies research*. 2015;2015.

119. Lee S, Talen E. Measuring Walkability: A Note on Auditing Methods. *Journal of Urban Design*. 2014;19(3):368-388.

120. Marshall WE, Garrick NW. Effect of street network design on walking and biking. *Transportation Research Record*. 2010;2198(1):103–115.

121. Brownson RC, Hoehner CM, Day K, Forsyth A, Sallis JF. Measuring the Built Environment for Physical Activity: State of the Science. *American Journal of Preventive Medicine*. 2009;36(4, Supplement):S99-S123.e12.

122. Clifton KJ, Smith ADL, Rodriguez D. The development and testing of an audit for the pedestrian environment. *Landscape and Urban Planning*. 2007;80(1-2):95–110.

123. Harvey CW. Measuring streetscape design for livability using spatial data and methods. 2014.

124. Charreire H, Mackenbach JD, Ouasti M, et al. Using remote sensing to define environmental characteristics related to physical activity and dietary behaviours: a systematic review (the SPOTLIGHT project). *Health & place*. 2014;25:1–9.

125. Dubey A, Naik N, Parikh D, Raskar R, Hidalgo CA. Deep Learning the City: Quantifying Urban Perception at a Global Scale. In: Leibe B, Matas J, Sebe N, Welling M, eds. *Computer Vision – ECCV 2016*. Cham: Springer International Publishing; 2016:196–212.

126. Gehl J. *Cities for People*. Island press; 2013.

127. NYC Department of Buildings. *Facades and Retaining Walls: Laws, Rules, and Filing Protocols*. https://www1.nyc.gov/assets/buildings/pdf/facades_retaining_walls.pdf.

128. United States Department of Justice Civil Rights DIvision. *Curb Ramps and Pedestrian Crossings Under Title II of the ADA*. https://www.ada.gov/pcatoolkit/ch6_toolkit.pdf.

129. Gothamist. *NYC Agrees To Make All Sidewalk Curbs Accessible To The Disabled*. http://gothamist.com/2019/03/21/sidewalk_curbs_accessible_nyc.php.