ISSN 2464-4617(print) ISSN 2464-4625(CD) CSRN 2701 Computer Science Research Notes
http://www.WSCG.eu

Predicting vehicle trajectories from surveillance video in a real scenario with Histogram of Oriented Gradient

Arthur Emidio T. Ferreira
University of Brasilia, Brazil
arthur.500@gmail.com

Bruno Luiggi M. Espinoza
University of Brasilia, Brazil
bruno@cic.unb.br

Flavio de Barros Vidal
University of Brasilia, Brazil
fbvidal@unb.br

Abstract

We propose a method capable to predict vehicle trajectories in a real scenario based on an unsupervised approach using Histogram of Oriented Gradients (HOG) features to construct an uniform path. The proposed algorithm extracts a sub-region of the input image defined as Field of View of the target vehicle, to output a possible trajectory that the given vehicle will follow through. We perform many experiments using the proposed technique, and based on qualitative/quantitative analyses, we conclude it is successfully able to predict reasonable trajectories.

Keywords

Histogram of Oriented Gradients, Path Prediction and Planning, Trajectory Forecasting.

1 INTRODUCTION

Consider the scene described in Figure 1. We, as humans, are able to predict the trajectory that the highlighted vehicle is likely to traverse in order to reach the goal indicated by the red circle. This ability is possible due to our capacity of using prior knowledge to forecast visual events [CBM12]. For instance, we may infer that in order to reach the destination, the vehicle won't collide with any other cars or pedestrians, nor have any contact with the sidewalk.

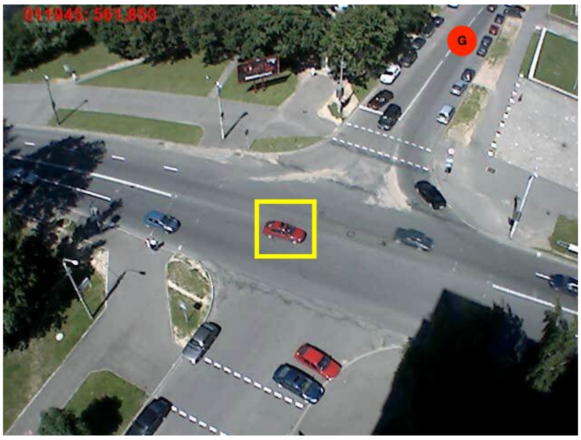

Figure 1: Given the highlighted vehicle and a goal point. What would be the trajectory traversed by the car before reaching the destination?

In computer vision, the topic of trajectory prediction has been explored in recent works, as presented in [WJM14, YY3, HSZ16], with the goal to forecast the trajectories of active elements in a static input image. A characteristic, that these works share in common, is the use of training data in order to give an output. In [HSZ16], for example, uses deep learning techniques to compute the most possible paths that an active agent is likely to traverse.

Trajectory forecasting is a very important topic in computer vision. For instance, predicting paths can improve the effectiveness of object tracking algorithms when dealing with significant occlusions [FBV16]. Moreover, prediction takes a very important role in scene understanding.

Is it possible to predict trajectories using a method that is not based on training data? In this work, we attempt to answer such question by proposing a framework that uses HOG features of the input image to compute future paths traversed by vehicles in a road.

The gradient of an image is an interesting approach to extract information about texture [ZZS14]. Based on previous knowledge, we know that vehicles tend to move on roads, a surface that is usually uniform in terms of texture. Therefore, we can compare HOG [DTB05] blocks to analyse different trajectories (sharing the same start and goal points) to determine which one is more suitable for a moving vehicle.

The previously mentioned works in trajectory forecasting propose generalized frameworks of prediction, meaning that it should be able to predict various types of active agents in different scenarios (e.g. pedestrians), based on what the training data consists of. In our proposed work, we restrict the domain of application because the characteristic of trajectories on uniform textured surfaces is inherent in vehicles moving on a road. However, since our method is primarily based on HOG features, which are very robust to be computed, then the time and space efficiency can be greatly increased, meaning that our approach can provide prediction results significantly faster than other approaches, since ours doesn't rely on training.

This paper is organized as follows: Section 2 provides a brief background knowledge regarding event forecasting; Section 3 presents the proposed approach applied for path prediction; Section 4 presents the application and validation of the proposal in a specific vehicle path forecasting; and final remarks and future work envisioned are provided in Section 5.

2 RELATED WORKS

Prediction is an inherent human ability [SK10a], and has also been observed in several animal species [RW14], such as in Western scrub-jays [CSP07]. Event prediction is an important trait to comprehend and respond to the environment. Therefore, various recent works have been developed with the goal to bring such skill to the field of computer vision.

In computer vision, prediction has started to be explored in recent years for: i) tracking occluded objects [FHS10], ii) predicting missing frames or extrapolating future frames in a video [RMB14], iii) semantically forecasting the future contextual events that are likely to happen in unlabeled videos [VCP15], iv) predicting the future motion of individual pixels in a static image [MH16], v) anticipating human events so robots can better assist humans in daily activities [KS13], and vi) predicting the consequences of forces applied to objects in images [MR16].

We explore the topic of path prediction. Several techniques have been developed to predict the trajectories of active agents in a given scene. In the current literature, there are many works to predict: trajectories in egocentric videos [SKK16], the most likely trajectories of players in football games [LN16], predict trajectories performed by pedestrians [KKZ12, PS11], and more general approaches [WJM14, YY3, HSZ16].

The work done by [WJM14] consists of an unsupervised technique to predict the most possible trajectories that an active agent is likely to follow in a scene. The proposed method is based on the extraction of mid-level patches [SSG12] present in each frame. Then, it is created a transition matrix containing information about how each element can move or transition into another patch. It is also created a reward function for each element, which maps how likely a patch can move to any point in space. The problem of prediction is thus solved by modelling a graph, where each node is a state (i.e. a patch located in a 2-dimensional point in the image), and each edge corresponds to the transition from one state to another, weighted based on the information present in the transition matrix and the reward function. This turns into an maximization problem: find the sequence of states that maximizes the reward function, where the goal states are along the edges of the image. The author solves this problem using Dijkstra's algorithm [CTH2]. The work shows results of this technique using datasets of vehicles and pedestrians.

One limitation of the work of [WJM14] is that the proposed method does not take the movement of co-occurring elements into consideration. However, the work presented in [YY3] uses a Kanade-Lucas-Tomasi (KLT) trackers [SJ94] to obtain object trajectories, where each trajectory is converted into a quantized form. The work proposes an unsupervised Hierarchical Topic-Gaussian Mixture Model (HTGMM) that extracts semantical movement patterns (e.g. go straight, turn left) based on quantized trajectories. These patterns are divided into groups, such that all movement patterns inside a group may occur simultaneously. Based on this information, an energy potential map is created and iteratively updated, allowing to predict future trajectories considering the movement of other agents in scene.

Another recent work in the field of path prediction is presented in [HSZ16]. It proposes a framework that uses deep learning to predict future trajectories. The proposed method is based on two CNNs: a Spatial Matching Network and an Orientation Network. The Spatial Matching Network is responsible for generating a reward map of the scene, which is used to check if an agent is likely to reach a given region. The Orientation Network is responsible for estimating the agent orientation in order to predict the most possible direction that the agent is likely to pursue in the near future. Based on the information present in the reward map and in the estimated orientation, the technique of [HSZ16] uses a unified path planning scheme to predict future trajectories.

Our method is based on the technique of Histogram of Oriented Gradients (HOG) [DTB05]. The HOG technique has been commonly used with Support Vector Machines to perform human and object detection [BH2014]. HOG has also been used to assist on some previous prediction techniques. In the work of [WJM14], each extracted patch is considered to be a HOG cluster. The work of [KT10] uses HOG as one of the image representations to predict what could be observed if the camera changed its position. Here, we propose a technique that considers HOG as the main element of the path prediction process.

One similarity between the methods proposed in the works of [WJM14, YY3, HSZ16] is that they use a previously training stage in order to forecast future trajectories. One of the reasons for doing this is to make predictions more accurate in various types of scenarios. Here, we propose a simpler framework to forecast vehicle trajectories. In this domain, we observe that vehicles tend to move in uniform regions (e.g. a vehicle usually maintains itself in a road area, which tends to be different in texture compared to a sidewalk, a pedestrian, or another vehicle). HOG can be a useful technique to find texture differences inside a small region. Therefore, we propose HOG to be used as a feature to detect regions that a vehicle is likely to move at. Additionally, the computational time of HOG features can be done in a very efficient manner, both in terms of time and space, therefore we propose a method that can be executed very close to real-time in today's hardware.

3 PROPOSED APPROACH

The objective behind our method is to predict the trajectory that a vehicle will follow to reach a given destination point using only the information from an image.

Therefore, we propose a framework (Figure 2) with the following input and output descriptions:

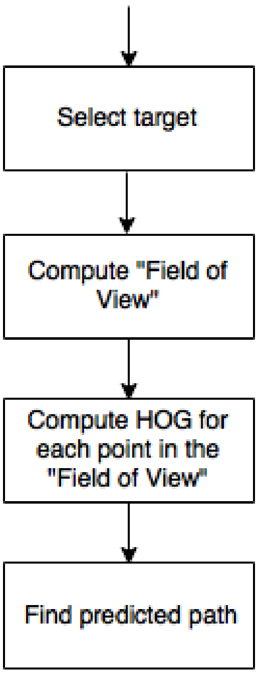

Figure 2: A flowchart describing the stages of the proposed approach.

Input: a small sequence of n sequential image frames $F = (f_1, f_2, ..., f_n)$ s.t. $n > 1$ taken by a stationary camera, the bounding rectangle R covering the vehicle that will have interest in its trajectory predicted in f_1, and a destination point D.

Output: a trajectory $T = (p_1, p_2, ..., p_m)$, where p_i is the i$^{\text{th}}$ two-dimensional point composing the path. T is the predicted trajectory for the input vehicle to reach the destination point D, based on frame f_1. The next four subsections explore each stage of the proposed algorithm and a visual representation of all these stages can be see in Figure 3.

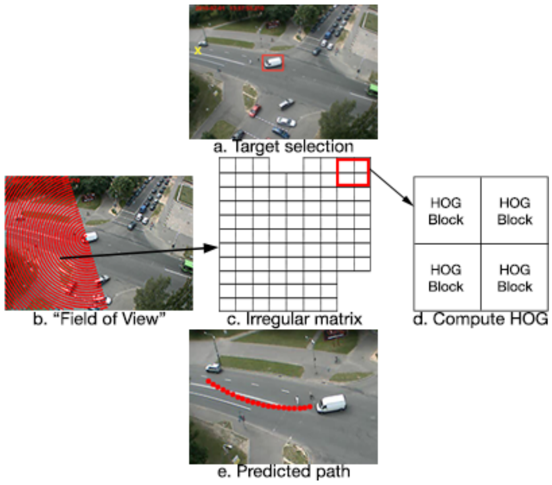

Figure 3: A visual representation of our proposed approach.

3.1 Target selection

The initial step of our approach is to provide the inputs from the image frame. First, we delimit a rectangle that covers the vehicle we are interested to have the trajectory predicted in frame f_1. Second, we mark the destination point D inside the boundaries of f_1.

3.2 Computing the "Field of View"

We denote the meaning of "Field of View" of a given vehicle as the set of all points that are located in the plane of 180^o created in terms of the orientation on which the vehicle is heading to. A visual representation of a field of view can be seen in Figure 3b.

We compute a "Field of View" to minimize the number of HOG cells to be computed in next step, with the goal of optimizing the average execution time of the algorithm.

The reason for using n input frames instead of only one is because we must infer in the selected vehicle's orientation. Therefore, we need a small number of n consecutive frames which should be enough to observe a noticeable motion of the vehicle between frames f_1 and f_n. Concurrently, n should not be too large, otherwise we could infer an incorrect vehicle's orientation.

Therefore, in order to find an orientation between frames f_1 and f_n, we must find the vehicle's location at frame f_n. The vehicle's location is achieved using template matching [FBV07] on frame f_2, having as a template image the *Region of Interest* (ROI) corresponding to the initial selection in the first step. Then, we update the template image, following the schema described in [FBV07], with the previous result from template matching, and repeat the process until the vehicle is located at frame f_n.

Let M_i be the 2-dimensional point of the center of mass on the selected vehicle at frame f_i. Thus, we can compute the orientation of the vehicle between frames f_1 and f_n by calculating the slope m of the line connecting M_1 and M_n. Based on the slope, we can obtain the orientation angle θ.

From θ, we can compute 180 parallel lines starting at M_1 and reaching the edges of frame f_1, in terms of the vehicle's direction. All points composing these 180 lines are defined as the "Field of View" of the vehicle with center of mass M_1 in f_1.

3.3 Computing HOG for each cell in the "Field of View"

In order to use the field of view in our prediction scheme, we must discretize it as an irregular matrix, which each element in the matrix corresponds to a point in the field of view (see Figure 3c). We do this using Algorithm 1.

```
Input
     I: current frame.
     D: the distance between two neighboring
        nodes sharing the same line in the
        field of view.
    VP: the vehicle's initial position point.
EndInput.

Output
     M: the irregular matrix.
    DV: a matrix storing the HOG descriptor
        values for each element in M.
EndOutput.

Begin
    For each ith line L in the Field of View:
        CP := VP
        j := 1
        While CP is in frame:
            M[i][j] := CP
            DV[i][j] := compute HOG in a 32x32
                        ROI of I st. the central
                        point is M[i][j].
            CP := point P in L
                  s.t. dist(P, CP) = D
            j := j + 1
        EndWhile.
    EndFor.
End.
```

Algorithm 1: Algorithm to compute HOG for each cell in the field of view

The next step is to calculate the HOG features for each element in the irregular matrix (Figure 3d). We do this by extracting a ROI of size 32×32 pixels defining the central point of this region as the point stored in the matrix element. Then, we compute the HOG features of the ROI using a HOG block of the same size as the ROI. The resulting vector $DV_{i,j}$, containing the descriptor values for each element in the irregular matrix of row i and column j, must be saved for future use.

3.4 Predicting the path

The last step of our approach consists in transforming the irregular matrix into a directed graph, where each vertex corresponds to an element of the irregular matrix, being adjacent to all vertices that are in its 8-neighborhood region in the irregular matrix.

We assign a cost $c_{i,j}$ for every edge connecting the vertex v_i to v_j. In order to show how it is calculated, let H_x be the resulting HOG features vector for node v_x, which was obtained in the previous step. Consider the following equation:

$$|H_i - H_j| = \left| \begin{bmatrix} h_{i_1} \\ h_{i_2} \\ \vdots \\ h_{i_m} \end{bmatrix} - \begin{bmatrix} h_{j_1} \\ h_{j_2} \\ \vdots \\ h_{j_m} \end{bmatrix} \right| = \begin{bmatrix} |h_{i_1} - h_{j_1}| \\ |h_{i_2} - h_{j_2}| \\ \vdots \\ |h_{i_m} - h_{j_m}| \end{bmatrix} \quad (1)$$

Based on Equation 1, we define the cost $c_{i,j}$ in Equation 2.

$$c_{i,j} = 1^T |H_i - H_j| = |h_{i_1} - h_{j_1}| + \ldots + |h_{i_m} - h_{j_m}| \quad (2)$$

Finally, we use the $A*$ search algorithm to compute the predicted path of our proposed framework [PNR68]. However, we must first set the initial and goal vertices. Let the initial vertex be the one corresponding to the first element in the 90th line of the field of view. Additionally, let the goal vertex be the one which is nearest to the destination point D which was selected as input in the first step of our approach.

The optimal path given by the $A*$ algorithm is the result of our prediction model, which can be seen in Figure 3e.

4 EXPERIMENTAL RESULTS

In order to demonstrate the effectiveness of our proposed methodology, we performed many experiments with several frames obtained from a image dataset, and compared the predicted trajectory with a ground truth achieved manually from the image sequence. We analysed the results in qualitative and quantitative manners. In the next subsections we describe the dataset, and present and discuss our results.

4.1 Dataset

For our experiments, we were interested in using a dataset that would contain several elements apart from vehicles and roads, such as pedestrians, sidewalks, trees, and different types of terrain (e.g. grass). We found all these elements of interest in the Minsk dataset [SNA14]. The Minsk dataset consists in a collection of video recordings from four different angles. In one of the angles, we are able to see a road intersection in aerial view. We used this subset as our primary experimental data presented in this work, given that crossroads present many possibilities of vehicle movements, and usually include more obstacles in scene.

4.2 Experiments

We present the result of two different input images from the Minsk dataset, which can be seen in Figures 4 to 9.

Figure 4: Exp. 1: The selected vehicle (in yellow) and its goal (in red).

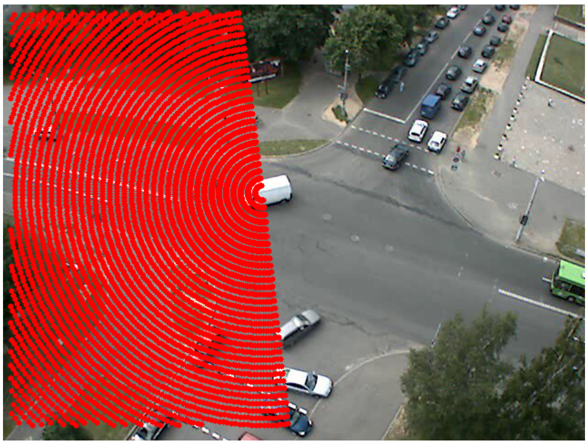

Figure 5: Exp. 1: The field of view of the selected vehicle.

For both experiments, we used the following parameters for the computation of HOG features:

- Block size: 16x16 pixels.

Figure 6: Exp. 1: The predicted trajectory of the selected vehicle.

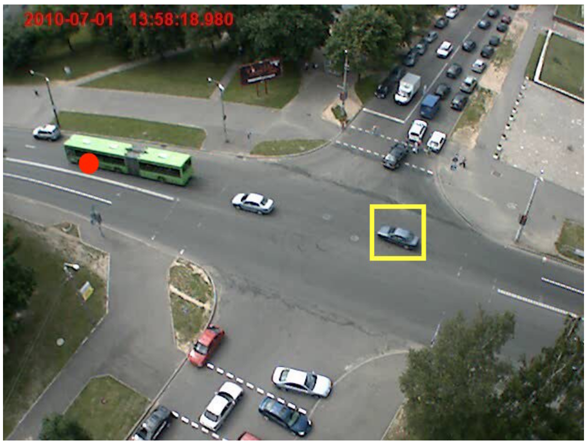

Figure 7: Exp. 2: The selected vehicle (in yellow) and its goal (in red).

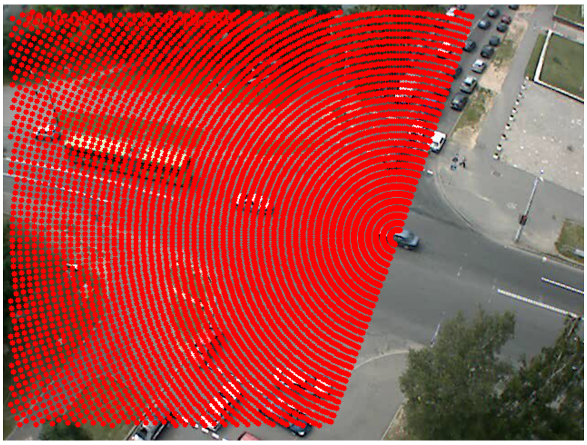

Figure 8: Exp. 2: The field of view of the selected vehicle.

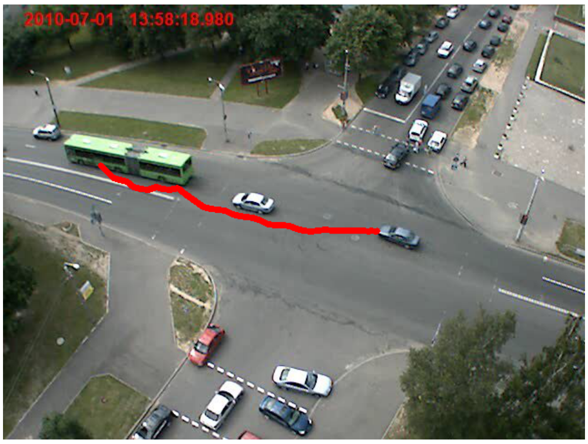

Figure 9: Exp. 2: The predicted trajectory of the selected vehicle.

- Cell size: 8x8 pixels.
- Number of bins: 9.
- Distance between blocks (in terms of its centroid): 8 pixels.

The Field of View from all experiments were obtained using a total of 5 (five) consecutive frames from the video sequence. Moreover, the goal from all experiments are the real vehicle's destination point, obtained from the ground truth. We choose two experiment scenarios, as described below.

4.3 Experiment 1

First, we can observe in Figure 5 that the field of view matches the direction that the vehicle is pointed towards to. Second, we can visualize the predicted trajectory in Figure 6. The path successfully avoids any contact with the pedestrians that are close to the selected vehicle. It is also observable that a considerable part of the trajectory remains in the road's center line, this can be explained due to the uniformity that this region tends to have, therefore, the absolute difference between HOG blocks in these parts of the frame are minimal.

4.4 Experiment 2

In this example, we select a goal that is farther away compared to the first experiment. Hence, the field of view, seen in Figure 8, is considerable larger, but still coinciding with the vehicle's direction. In the predicted trajectory, seen in Figure 6, we can observe that the vehicle is capable in avoiding a collision with the white car located in its front. Given that the goal was selected to be at where the bus is located, a collision is unavoidable. However, it can be seen that such collision only happens very close to the goal point.

After describing the experiments scenarios, for a quantitative analysis, we ran our technique for all frames that the selected vehicle is present in scene, and computed the average error between the predicted trajectory and the path obtained from the ground truth. We used the Equation 3 to compute the error for iteration i:

$$E_i = \frac{\sum_{j=1}^{n} minDist(B_j, GT)}{n}, \qquad (3)$$

where n is the number of points in the predicted trajectory, B_j is the jth point of the predicted path, GT is the set of points in the ground truth, and the function $minDist(x,Y)$ computes the euclidean distance from point x to the its nearest neighbor in the set of points Y.

We present the mean errors in Figures 10 and 11 using the selected vehicles from experiments 1 and 2.

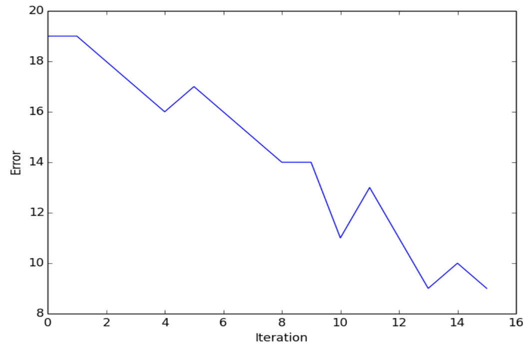

Figure 10: The errors calculated using the vehicle from Experiment 1.

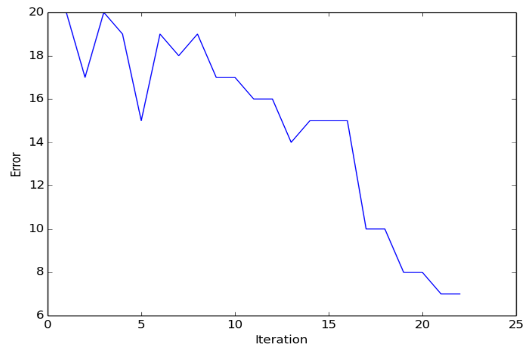

Figure 11: The average errors calculated using the vehicle from Experiment 2.

As we can observe in both plots, the mean error tends to decrease in each iteration. This can be explained due to the fact that the vehicle's field of view is smaller, and consequently the number of possible trajectories are decreased. Therefore, our method is expected to provide more accurate predictions when the vehicle is closer to its goal.

5 CONCLUSION

We have proposed a simple and efficient method to predict vehicle trajectories in a real scene using the Histogram of Oriented Gradients as the main feature and optimization algorithms. Based on the assumption that vehicles are likely to move in uniform regions (i.e. avoiding obstacles and different types of terrain), we have shown that the comparison between HOG descriptor vectors is an interesting approach to find differences in texture between small adjacent regions. We have demonstrated that our proposed framework provides satisfactory results with aerial view videos, being able to propose paths that don't intersect with additional vehicles, pedestrians, and other non-asphalt regions.

Additionally, given the time complexity $O(|E|)$ of the $A*$ search algorithm, and considering that the computation of HOG descriptors can be performed considerably fast in today's hardware, we have shown that the time/space efficiency of predicting vehicle trajectories is considerably positive and achievable for any computer architecture, including the mobile and embedded devices.

However, it is important to remark that our proposed method is an initial step in trajectory prediction in video. For future work, we plan to be able to predict trajectories of a vehicle by considering the existence of changes in the environment (e.g. movement of other vehicles and pedestrians). Additionally, we plan to use HOG as a feature in a trained model in order to observe whether we are able to forecast better trajectories. By using a trained model, we also plan to predict the vehicle's final position. Furthermore, since our goal is to predict paths executed by vehicles, it is very important to construct a framework that infers common traffic laws, such as recognizing whether a street only allows one-way traffic.

6 REFERENCES

[BH2014] Bristow, Hilton, and Lucey, Simon. 2014. *Why do linear SVMs trained on HOG features perform so well?* CoRR, abs/1406.2419.

[CBM12] Cheung, Olivia S, & Bar, Moshe. 2012. Visual prediction and perceptual expertise. *Int J Psychophysiol*, **83**(2), 156–63.

[CTH2] Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. Dijkstra's algorithm. *Chap. 24, pages 595–599 of: Introduction to Algorithms 2nd edition*. MIT Press.

[CSP07] Correia, Sérgio P.C., Dickinson, Anthony, & Clayton, Nicola S. 2007. Western Scrub-Jays Anticipate Future Needs Independently of Their Current Motivational State. *Current Biology*, **17**(10), 856 – 861.

[DTB05] Dalal, N., & Triggs, B. 2005 (June). Histograms of oriented gradients for human detection. *Pages 886–893 vol. 1 of: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, vol. 1.

[FBV16] de Barros Vidal, Flavio, Koike, Carla M. C. C., Cordoba, Diego A. L., & Zaghetto, Alexandre. 2014. Improving Visual Tracking Robustness in Cluttered and Occluded Environments using Particle Filter with Hybrid Resampling. *Pages 605–612 of: Proceedings of the 9th International Conference on Computer Vision Theory and Applications (VISIGRAPP 2014)*.

[FHS10] Fu, Hui-Xuan, Sun, Feng, & Liu, Sheng. 2010 (July). Anti-occlusion tracking algorithm based on LSSVM prediction and Kalman-MeanShift. *Pages 6031–6036 of: Intelligent Control and Automation (WCICA), 2010 8th World Congress on*.

[HSZ16] Huang, Siyu, Li, Xi, Zhang, Zhongfei (Mark), He, Zhouzhou, Wu, Fei, Liu, Wei, Tang, Jinhui, & Zhuang, Yueting. 2016. Deep Learning Driven Visual Path Prediction from a Single Image. *CoRR*, **abs/1601.07265**.

[KT10] Kaneva, B., Sivic, J., Torralba, A., Avidan, S., & Freeman, W. T. 2010. Matching and Predicting Street Level Images. *In: ECCV 2010 Workshop on Vision for Cognitive Tasks*.

[KKZ12] Kitani, Kris M., Ziebart, Brian D., Bagnell, James Andrew, & Hebert, Martial. 2012. *Activity Forecasting*. Berlin, Heidelberg: Springer Berlin Heidelberg. Pages 201–214.

[KS13] Koppula, H. S., & Saxena, A. 2013 (Nov). Anticipating human activities for reactive robotic response. *Pages 2071–2071 of: 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*.

[LN16] Lee, Namhoon, & Kitani, Kris M. 2016. Predicting wide receiver trajectories in American football. *Pages 1–9 of: 2016 IEEE Winter Conference on Applications of Computer Vision, WACV 2016, Lake Placid, NY, USA, March 7-10, 2016*.

[MR16] Mottaghi, Roozbeh, Rastegari, Mohammad, Gupta, Abhinav, & Farhadi, Ali. 2016. "What happens if..." Learning to Predict the Effect of Forces in Images. *CoRR*, **abs/1603.05600**.

[PNR68] P. E. Hart, N. J. Nilsson, & Raphael, B. 1968. A formal basis for the heuristic determination of minimum cost paths. *IEEE Transactions on Systems, Science, and Cybernetics*, **SSC-4**(2), 100–107.

[PS11] Pellegrini, Stefano, Ess, Andreas, & Van Gool, Luc. 2011. *Predicting Pedestrian Trajectories*. London: Springer London. Pages 473–491.

[RMB14] Ranzato, Marc'Aurelio, Szlam, Arthur, Bruna, Joan, Mathieu, Michaël, Collobert, Ronan, & Chopra, Sumit. 2014. Video (language) modeling: a baseline for generative models of natural videos. *CoRR*, **abs/1412.6604**.

[RW14] Roberts, William A. 2012. Evidence for future cognition in animals. *Learning and Motivation*, **43**(4), 169 – 180. Remembering the Future: The Influence of Past Experience on Future Behavior.

[SNA14] Saunier, Nicolas, Ardo, Hakan, Jodoin, Jean-Philippe, Nilsson, Aliaksei Laureshyn Mikael, Svensson, Åse, Miranda-Moreno, Luis, Bilodeau, Guillaume-Alexandre, & Astrom, Kalle. 2014. *A Public Video Dataset for Road Transportation Applications*.

[SJ94] Shi, Jianbo, & Tomasi, Carlo. 1994. Good Features to Track. *Pages 593 – 600 of: 1994 IEEE Conference on Computer Vision and Pattern Recognition (CVPR'94)*.

[SKK16] Singh, Krishna Kumar, Fatahalian, Kayvon, & Efros, Alexei A. 2016. KrishnaCam: Using a longitudinal, single-person, egocentric dataset for scene understanding tasks. *Pages 1–9 of: 2016 IEEE Winter Conference on Applications of Computer Vision, WACV 2016, Lake Placid, NY, USA, March 7-10, 2016*.

[SSG12] Singh, Saurabh, Gupta, Abhinav, & Efros, Alexei A. 2012. Unsupervised Discovery of Mid-level Discriminative Patches. *In: European Conference on Computer Vision*.

[SK10a] Szpunar, Karl K. 2010. Episodic Future Thought: An Emerging Concept. *Perspectives on Psychological Science*, **5**(2), 142–162.

[FBV07] Vidal, F. B., & Alcalde, V. H. C. 2007 (June). Object Tracking by introducing Stochastic Filtering into Window-Matching Techniques. *Pages 31–36 of: 2007 International Symposium on Computational Intelligence in Robotics and Automation*.

[VCP15] Vondrick, Carl, Pirsiavash, Hamed, & Torralba, Antonio. 2015. Anticipating the future by watching unlabeled video. *CoRR*, **abs/1504.08023**.

[WJM14] Walker, Jacob, Gupta , Abhinav, & Hebert , Martial. 2014 (March). Patch to the Future: Unsupervised Visual Prediction. *In: Proc. Computer Vision and Pattern Recognition*.

[MH16] Walker, Jacob, Doersch, Carl, Gupta, Abhinav, & Hebert, Martial. 2016. An Uncertain Future: Forecasting from Static Images using Variational Autoencoders. *CoRR*, **abs/1606.07873**.

[YY3] Yoo, YoungJoon, Yun, Kimin, Yun, Sangdoo, Hong, JongHee, Jeong, Hawook, & Young Choi, Jin. 2016 (June). Visual Path Prediction in Complex Scenes With Crowded Moving Objects. *In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[ZZS14] Zuo, W., Zhang, L., Song, C., Zhang, D., & Gao, H. 2014. Gradient Histogram Estimation and Preservation for Texture Enhanced Image Denoising. *IEEE Transactions on Image Processing*, **23**(6), 2459–2472.