ESANN 2017 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 26-28 April 2017, i6doc.com publ., ISBN 978-287587039-1.
Available from http://www.i6doc.com/en/.

Supporting Generative Models of Spatial Behavior by User Interaction

Ronny Hug, Wolfgang Hübner and Michael Arens

Fraunhofer Institute of Optronics, System Technologies and Image Exploitation
Gutleuthausstraße 1, 76275 Ettlingen

Abstract. The analysis of spatial behavior in terms of motion profiles recorded along trajectories is a widely used technique in video analysis. Inherent to this approach is the problem to assign a meaningful score to observations. This score builds the basis for classification, ranking, or to generate user feedback. Score assignment can be done in terms of deviations from normal behavior, where normality is determined by learning a generative model. A general drawback is that the unsupervised learning process often assigns non-intuitive scores. In order to address this problem this paper proposes the usage of interactive concepts, which support the learning process. Interaction thereby strongly utilizes the generative models capabilities to synthesize samples, to give insight into the underlying representation. Initial results are shown on a trajectory rating task, illustrating the feasibility of the proposed approach.

1 Introduction

The task of assigning meaningful scores to motion profiles arises in different contexts, like video based situation assessment or visual log file generation (e.g. [1]). This score assignment task can be addressed by using a generative model, directly learned from observed motion profiles. Based on the assumption that usual behavior occurs more frequently, the model automatically learns to rank motion profiles based on what is assumed to be normal. However, in practice, the observation time of a system can be quite long, whereby misleading data may be captured. Additionally, not every outlier is a critical event. These issues can be addressed by incorporating user interaction into the unsupervised learning process, which is then used to guide the process and to fine-tune the model.

Interactive approaches are already successful in different tasks like recommender systems[2][3], search[4][5][6] or clustering[7][8]. These approaches work by either asking the user or by observing the users activities, in order to shape a learned model to fit the users needs. Hence, the user provides examples to the learner, thus guiding the learning process. This paper additionally proposes a method of incorporating user feedback concerning outliers generated by the learned model. In this manner the decision boundary of the model can be manipulated interactively, without the need for more data.

Finally, this paper proposes a novel interactive approach for learning generative models of spatial behavior, especially motion profiles. The following sections give an overview over the score assignment problem and the proposed approach. The feasibility of the approach is illustrated on a trajectory rating

ESANN 2017 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 26-28 April 2017, i6doc.com publ., ISBN 978-287587039-1.
Available from http://www.i6doc.com/en/.

task by providing a set of different interaction concepts for modeling motion along trajectories using the BIWI Walking Pedestrians dataset[9].

2 Learning the Generative Model with User Interaction

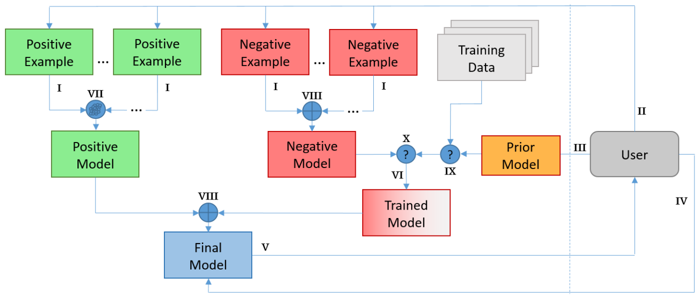

Fig. 1: Interactive Learning Architecture. I: User-defined Parameters, II: Chosen by User, III: User-defined Prior, IV: User Outlier Feedback, V: Scoring and Visualization of Data and Outliers, VI: Variational EM-like Algorithm, VII: Fusion by Sampling, VIII: Aligning Fusion, IX: Positive Filter, X: Negative Filter.

This section gives an overview over the trajectory score assignment problem and the proposed interactive learning process. The process is designed to allow a user to influence the training steps and to fine-tune the learned representation. An overview of the architecture is depicted in figure 1.

Motion Modeling. Motion profiles are usually described as a sequence of time-discrete measurements (e.g. positions) obtained via object tracking algorithms. As this representation is not descriptive enough for generating reasonable synthetic trajectories, each measured position $\mathbf{p_t}$ is extended by the *shift* $\delta_\mathbf{t} = \mathbf{p_t} - \mathbf{p_{t-1}}$, thus introducing a dependency between subsequent positions. Assuming that each state only depends on its predecessor (*markov assumption*), a trajectory, consisting of points $\mathbf{x_t} = (\mathbf{p_t}, \delta_\mathbf{t})$, can be described as a path in a continuous-state markov chain. Therefore, a gaussian mixture model (GMM) over the joint distribution of point pairs $p(\mathbf{x_t}, \mathbf{x_{t-1}})$ is used to model motion between trajectory points. Using a GMM provides an easy way of calculating $p(\mathbf{x_t}|\mathbf{x_{t-1}})$, which resembles the transition kernel in the markov chain.

Score Assignment. One possibility of rating a trajectory is to calculate a score that resembles the grade of membership to the final model. In general, the probability of a path $X_N = (\mathbf{x_0}, ..., \mathbf{x_N})$ in a markov chain is defined as $p(X_N) = p(\mathbf{x_0}) \prod_{t=1}^{N} p(\mathbf{x_t}|\mathbf{x_{t-1}})$. To obtain a score $s \in (0, 1]$ from $p(X_N)$, the probability density function (pdf) values have to be scaled and the dependency on the chain length has to be considered. Therefore, the score is defined as $s = \frac{1}{N+1} p^s(\mathbf{x_0}) \prod_{t=1}^{N} p^s(\mathbf{x_t}|\mathbf{x_{t-1}})$, where $p^s(\mathbf{x}) = \frac{p(\mathbf{x})}{max_z p(\mathbf{z})}$.

ESANN 2017 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 26-28 April 2017, i6doc.com publ., ISBN 978-287587039-1.
Available from http://www.i6doc.com/en/.

Interaction Concepts. In contrast to black box learners, where all training data is used, or active learners, where an oracle is needed frequently, the proposed interactive learning process learns auxiliary models and filters from data to augment its training steps. In this case, the learning process is a continuous process, where a user can intervene at different stages. More precisely, the process starts by collecting data and then performs an initial unsupervised training step, which enables trajectory rating. This training step can be influenced by providing information about expected behavior (e.g. specific movement directions or speed). Afterwards, the user can issue training steps by providing positive or negative examples observed at run-time and corresponding variation parameters (e.g. variations in speed), which are collected to form additional stages in the learning process. In this way, the user directly influences the process and contributes to handling difficulties in training generative models, e.g. underfitting.

Additionally, the interactive learning process provides the possibility to interactively manipulate the final model directly via outlier assessment. For this purpose the generative model is used to synthesize samples at the boundary, which are then visualized. Currently, outliers are generated along each of the 4 dimensions: position (p_x, p_y) and shift (δ_x, δ_y). An outlier is generated pointwise by moving along one of the four axis until the score of the point (or transition) drops below a certain threshold. Given such an outlier, the user can decide to increase or decrease the score, thus adapting the model parameters and finally shifting the outlier boundary. By assigning a higher score to the outlier, the outlier boundary is pushed further *outwards*, resulting in outliers of the manipulated model being more different from a prototypical trajectory. The same principle applies for pushing *inwards* when assigning a lower score. This type of interactive learning can be seen as a fine-tuning step for the final model.

Architecture. To influence the training steps, the user is allowed to provide examples (or prototypical trajectories) and corresponding variation parameters. For each example a synthetic model is learned utilizing a variational EM-like algorithm[1] [10] and the variation parameters, where the example is a prototypical representative of that model. The models obtained in this way are fused into a positive, negative and a prior model (green, red and orange part in figure 1).

The prior model acts as a positive filter that selects a subset from the training data for training steps and can be defined by hand. To adapt the model at runtime, a positive and a negative model is built from examples provided by the user. These models are used in different ways: the positive model amplifies the trained model to get the final model and the negative model forms a negative filter for the next training step.

For fusing models (e.g. example models) two fusion strategies are used: *Fusion by Sampling* and *Aligning Fusion*. The former generates samples from each input model in order to train a new model, thus generally reducing the model complexity. The latter combines models and aligns their modes, thus increasing the model complexity.

[1] Also used for obtaining the trained model (see figure 1) from (filtered) data.

ESANN 2017 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 26-28 April 2017, i6doc.com publ., ISBN 978-287587039-1. Available from http://www.i6doc.com/en/.

3 Experiments

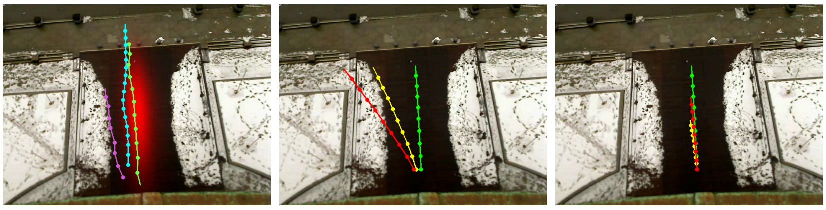

Fig. 2: ETH Zürich scene taken from the *BIWI Walking Pedestrians dataset*[9]. 1: Positional pdf for the used data subset (red area) and discrete trajectories used for interactive learning (purple: low score, green: high score, cyan: user-defined prior mean trajectory). 2,3: Feedback based on outlier representation manipulation (green: Prototype, yellow: outlier, red: outlier after manipulation).

Scores assigned to trajectories intuitively describe the grade of normality of the observed motion. By incorporating user interaction into learning and fine-tuning, a generative model should learn to cope with frequently observed, but abnormal, motion, and to distinguish between critical and non-critical outliers. Consequently, scores give an indication on how well the interactive learning approach performs in reaching these goals. To test the impact of user interactions on the score, the interactive learning process is benchmarked in different interaction scenarios. The benchmark utilizes a total of 61 trajectories (sequences of 3D positions on a plane, 60 of which taken from the *BIWI Walking Pedestrians dataset*[9]), where 58 are used for the initial training and the remaining three for interaction concepts. The data set consists of trajectories going from "top to bottom" (*tb*, 31+1 trajectories) and from "bottom to top" (*bt*, 27+2 trajectories). The first image in figure 2 shows a positional pdf of all trajectories. Further, a trajectory receiving a low score after an initial training without prior information (purple), a trajectory receiving a high score (green) and a hand-drawn trajectory that should be incorporated as prior information (cyan) are shown. The purple and green trajectories are assumed to be observed after the initial training step, thus these are left out in this step.

Manipulation via examples and prior information. Adding examples or prior information should show a direct impact on the scores assigned by the model to a given set of trajectories when compared to the scores assigned without interactive manipulation. Four experiments were run for evaluation (results in figure 3). The left diagram shows the mean scores of *bt* and *tb* trajectories after the initial training without prior information (I) and after additional application of different interactions (II - V). The right diagram shows the scores of the two provided example trajectories before and after the interaction.

In the first experiment (II, A), a low-scored *bt* trajectory is added as a positive example with low variation. As positive examples increase the model complexity,

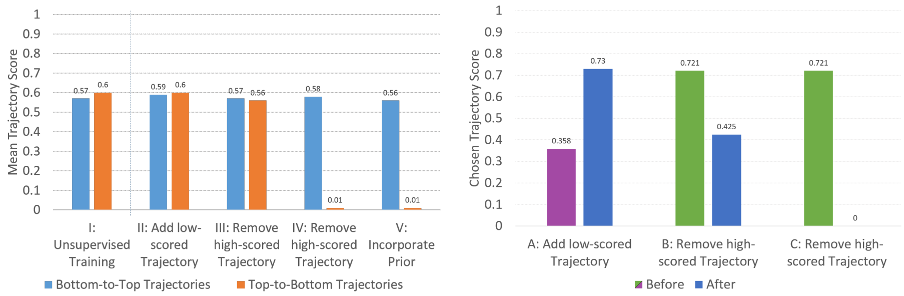

Fig. 3: Trajectory Scores before and after interactive training. I: Mean score of bt/tb trajectories without interactive training (reference score). II-V: Mean score of bt/tb trajectories with different interactions. A-C: Scores of trajectories chosen for interactive learning before and after interaction. IV/C uses higher variance than III/B.

the final model shows an amplification in the examples region. Therefore, only the scores of the example and similar trajectories are affected by this interaction. Additionally, this example is closer to the models boundary than to the center. Thus it won't affect most of the training trajectories and there is only a slight change in the mean scores, but a large change in the examples score.

In the second and third experiment (III, IV, B, C) a high-scored *tb* trajectory has been added as a negative example with low and high variation, respectively. Regarding the examples high score, it causes many trajectories to be filtered out, even with low variation. A low variation causes about half of the *tb* trajectories to receive a lower score, thus reducing the mean score. Nonetheless, there are enough *tb* trajectories akin the example that are not filtered out, which cause the "gap" produced by the filter to be closed in the training step. As a consequence even the filtered trajectories reach a score significantly above zero and finally the mean score only changes slightly. Choosing a higher variation for this example shows a stronger impact on the scores, as all *tb* trajectories are filtered out. As a result, the mean score of *tb* trajectories drops to a value near zero.

The last experiment (V) introduces a user-defined prior. For this example, a prototypical *bt* trajectory with an average walking speed of 1m/s was created by hand. By providing prior information, only trajectories that are similar enough to the provided prototype (*bt* trajectories in this case) are selected for training, leaving all others at a score near zero.

Manipulation via outlier assessment. Besides the model manipulation via observations, the possibility of manipulating the model directly via outliers is provided. In figure 2 a positional outlier (second image, yellow) and a shift outlier (third image, yellow) were generated, where in this case a positional outlier moves at the edge of the previously observed positions and a shift outlier moves too slow. The green trajectory shows a prototypical trajectory for the current model (using the model with prior information as described before). By assigning a higher score to the positional outlier, the outlier boundary is pushed further outwards, resulting in the outlier of the manipulated model (red trajectory) being further away from the prototype. The same principle applies to the shift outlier. When reducing its score, the outlier boundary is moved closer to the prototype, thus the movement speed of the new outlier is closer to the prototype.

4 Conclusions and Future Work

An approach for interactively learning and refining a generative model of motion profiles has been proposed. It consists of several interacting models, allowing a user to influence training steps via examples and prior information and to manipulate the learned model via outlier assessment. Experiments show, that the approach allows intuitive interactions that have a well defined impact on the generative model.

Future work will include investigating other generative models and their suitability for modeling, sample generation and inference, as the approach is not restricted to gaussian mixture models. Additionally more concepts for interactions and outlier generation will be elaborated. A long-term goal is to build an interactive framework for learning abstract concepts and human interactions based on spatial information.

References

[1] D. Münch, B. Hilsenbeck, H. Kieritz, S. Becker, A.-K. Grosselfinger, W. Hübner, and M. Arens. Detection of Infrastructure Manipulation with Knowledge-based Video Surveillance. In *Proceedings of SPIE Optics and Photonics for Counterterrorism, Crime Fighting and Defence 2016*, 2016.

[2] Negar Hariri, Bamshad Mobasher, and Robin Burke. Adapting to user preference changes in interactive recommendation. In *Proceedings of the 24th International Conference on Artificial Intelligence*, pages 4268–4274. AAAI Press, 2015.

[3] Svetlin Bostandjiev, John O'Donovan, and Tobias Höllerer. Tasteweights: a visual interactive hybrid recommender system. In *Proceedings of the sixth ACM conference on Recommender systems*, pages 35–42. ACM, 2012.

[4] Philippe Henri Gosselin and Matthieu Cord. Active learning methods for interactive image retrieval. *IEEE Transactions on Image Processing*, 17(7):1200–1211, 2008.

[5] Kan Ren, Risto Sarvas, and Janko Ćalić. Interactive search and browsing interface for large-scale visual repositories. *Multimedia Tools and Applications*, 49(3):513–528, 2010.

[6] Thomas S Huang, Charlie K Dagli, Shyamsundar Rajaram, Edward Y Chang, Michael I Mandel, Graham E Poliner, and Daniel PW Ellis. Active learning for interactive multimedia retrieval. *Proceedings of the IEEE*, 96(4):648–667, 2008.

[7] Sharad Vikram and Sanjoy Dasgupta. Interactive bayesian hierarchical clustering. *arXiv preprint arXiv:1602.03258*, 2016.

[8] Pranjal Awasthi, Maria-Florina Balcan, and Konstantin Voevodski. Local algorithms for interactive clustering. In *ICML*, pages 550–558, 2014.

[9] ETH Zürich Computer Vision Lab. BIWI Walking Pedestrians dataset. http://www.vision.ee.ethz.ch/en/datasets/. Accessed: 2016-10-12.

[10] M Bishop Christopher. Pattern recognition and machine learning. *Company New York*, 16(4):049901, 2006.