A Behavior-Based Population Tracker Can Parse Aggregate Measurements to Differentiate Agents

Dr. Jason H. Rife
Dept. of Mechanical Engineering
Tufts University
Medford, MA
jason.rife@tufts.edu

Dr. Samarth Swarup
Biocomplexity Institute & Initiative
University of Virginia
Charlottesville, VA
ss7rs@virginia.edu

Dr. Nasim Uddin
Dept. of Civil, Construction & Environmental Engineering
University of Alabama
Birmingham, AL
nuddin@uab.edu

***Abstract*—Closed-loop state estimators that track the movements and behaviors of large-scale populations have significant potential to benefit emergency teams during the critical early stages of disaster response. Such population trackers could enable insight about the population even where few direct measurements are available. In concept, a population tracker might be realized using a Bayesian estimation framework to fuse agent-based models of human movement and behavior with sparse sensing, such as a small set of cameras providing population counts at specific locations. We describe a simple proof-of-concept for such an estimator by applying a particle-filter to synthetic sensor data generated from a small simulated environment. An interesting result is that behavioral models embedded in the particle filter make it possible to distinguish among simulated agents, even when the only available sensor data are aggregate population counts at specific locations.**

Keywords- Bayesian Estimation, Disaster Response, Particle Filter, Synthetic Population

I. INTRODUCTION

Large-scale population simulations have demonstrated utility for analysis and optimization of disaster response. As one example, an agent-based simulation was implemented to prepare for a hypothetical terrorist attack in Washington D.C. [1]-[3]. This simulation predicted the behaviors and movements of approximately one million individual agents, roughly the population of the affected area. Each simulated agent was initialized using US Census and other demographic data, and agent behaviors were evolved in time to reflect sheltering in place, escape, attempts to reunite with family, or falling victim to panic. The simulation enabled different first-responder strategies to be assessed; results indicated that one of the most effective means of saving lives would be to re-establish communication quickly using, for instance, portable cellular-phone towers [4]-[6].

A natural extension is to fuse a dynamic population simulation with real-time sensing, in order to form a state estimator that would model the movements and behaviors of an actual population, for example, during the time immediately following a disaster event. State estimators provide a capability to filter sensor noise and to infer information that cannot directly be sensed [7]. Such inferences could be leveraged to redeploy first-responders to areas of greatest need, to generate tailored response strategies for specific neighborhoods depending on the nature of the disaster, or to provide rapid testing and evaluation of high-risk, high-reward response strategies. As compared with existing disaster-response simulations, implemented to run in advance of a disaster (i.e. as *open-loop* simulations), the state-estimator would operate through closed-loop feedback, with real-time sensor data steering the simulation to reflect actual events as they unfold. Our goal is to develop closed-loop state estimators to enable inference about the locations and behaviors of a population during a disaster, even when direct sensing is not possible.

A broad range of state-estimation techniques have been developed in the research literature and deployed in practice [8]. Generally speaking, state-estimation methods can be divided into optimization (minimum-error) and Bayesian approaches [9]. Under certain conditions, the two approaches are equivalent [8]. Because of the strongly nonlinear form of sensing and dynamic models in disaster-response scenarios, we have opted to pursue a Bayesian formulation for our application [10], specifically a particle filter [11].

One of the most challenging aspects of working with population counting sensors is the data-association issue. The data association problem occurs in any multi-target tracking application with sensors that do not uniquely identify each target [12],[13]. Data association issues are particularly common in primary radar and video surveillance applications [14],[15]. Data association for multi-target tracking with a Bayesian estimator remains an open topic of research, though significant progress has been made to date [16],[17].

An alternative to solving the data association problem is to model population as a flow obeying conservation of mass. If dynamic models consider the density of population at any particular location, the resulting model is similar in form to Eulerian models originally developed for solving fluid mechanics problems [18]. In addition to removing the data association problem, such flow-based models can be implemented with great computational efficiency. As such, Eulerian simulations have been used successfully to develop closed-loop state estimators to infer flows in applications as diverse as highway traffic analysis, air traffic control, and building evacuation [19]-[21]. Despite their computational

978-1-7281-5092-5/19/$31.00 ©2019 IEEE

efficiency, Eulerian models have the disadvantage that they cannot explicitly model behavioral changes for individual members of the population, nor can they capture population-level emergent behaviors that result from the interaction of simple individual behaviors [22].

Because we believe that individual behaviors are highly significant to disaster response modeling, we have focused on individual models (more commonly called agent-based modeling, or ABM) rather than Eulerian flow models.

A significant research challenge remains in defining methods to solve large-scale data association when fusing individual agents in an ABM with aggregate sensor measurements (e.g., population counts provided by surveillance cameras). At first blush, the problem may even seem ill-defined, since it is difficult to intuit how aggregate measurements that contain no identity information might be used to perform Bayesian measurement updates for individual simulated agents. *After all, how can a population count be used to identify a specific individual?*

In recent work, we have proposed a new data association approach that considers likely combinations of agents at any particular sensor location; sampling these combinations allows for sensor updates without requiring definitive association [10]. Given that this sampling approach correlates agent movements over time, the approach effectively infers some information about specific individuals. Furthermore, as we explore in this paper, agents can be distinguished by agent-specific behavior models, integrated by the estimator. This result has important consequences for practical applications, consequences that may be beneficial (tailoring disaster response to individual needs) but that may also be concerning (invading privacy through individualized tracking).

The main focus of this paper is to characterize the mechanism by which our state estimator distinguishes among simulated agents given only aggregate sensor data, which is nominally de-identified. To this end, we take a simulation-based approach, generating and analyzing synthetic measurements to evaluate the limits of Bayesian inference under ideal conditions. The remainder of the paper is organized as follows. Next, we briefly summarize our Bayesian-estimator approach and our simulation setup. Subsequently, simulation results are described and analyzed, in order to explore the mechanisms by which behavior-based models can sift the aggregate data to classify subgroups within the agent population. A brief summary concludes the paper.

II. Methodology

A. Simulation Setup

For the purposes of exploring the identifiability of individual agents, we implemented a relatively simple simulation which represented the physical environment as a network consisting of 100 nodes. Each node represents a physical location (such as a home, workplace, or intersection). Nodes were randomly placed in the plane and connected to their four nearest neighbors. Within this network were placed 100 agents, initially scattered with uniform probability across the location network. Of these agents, 50 were singletons with no affiliation to other simulated agents, 20 were pairs affiliated to one other simulated agent, 18 were members of groups of three, and 12 were members of groups of four.

A ground truth simulation was generated by allowing the 100 agents to move within the location network. All singleton agents exhibited the same behavior, which was to move toward the nearest exit, transitioning one node per time step. Occasionally, agents would pause for a time step to rest. When agents arrived at an exit node, they would remain at that location. Groups exhibited a slightly different behavior. Each group began by meeting at a specific rendezvous location. Once all members of the group reached the rendezvous site, the group proceeded together to the nearest exit node. Across the network of all 100 nodes, only 3 nodes were specified as allowable rendezvous points, and only 2 nodes as exits.

Synthetic sensor measurements (aggregate population counts at specific nodes) were generated from the ground truth simulation. The sensor noise model did not allow for false alarms; however, missed detections were modeled with a binomial distribution, assuming a 10% missed-detection probability per individual present at the sensor location. Altogether, eight sensors were considered, all placed at distinct locations (nodes) within the simulated environment.

Synthetic sensor data were used to steer the Bayesian state estimator. The Bayesian estimator was implemented as a particle filter with 50 particles per agent, totaling 5000 particles in all. Particle distributions were propagated in time through an ABM, as described in the next section, and corrected at each time step using the synthetic sensor data.

B. Bayesian Estimation Methodology

As with any Bayesian estimator, the primary steps of the algorithm are predication and correction. The prediction step propagates the state estimate $\hat{\mathbf{x}}$ from one time step to the next.

$$\underbrace{p(\hat{\mathbf{x}}_k)}_{\text{prior}} = \iiint \underbrace{p(\hat{\mathbf{x}}_k|\hat{\mathbf{x}}_{k-1})}_{\text{process noise}} \underbrace{p(\hat{\mathbf{x}}_{k-1}|\mathbf{y}_{k-1})}_{\text{posterior}} dV \tag{1}$$

At each time step k, the correction step weights the output of the prediction according to the likelihood of generating the observed measurement vector $\mathbf{y}_k$. The result is an updated (posterior) estimate of the state vector $\hat{\mathbf{x}}_k$ that reflects the sensor data.

$$\underbrace{p(\hat{\mathbf{x}}_k|\mathbf{y}_k)}_{\text{posterior}} = C \underbrace{p(\mathbf{y}_k|\hat{\mathbf{x}}_k)}_{\text{sensor noise}} \underbrace{p(\hat{\mathbf{x}}_k)}_{\text{prior}} \tag{2}$$

In our estimator, the vector of estimated states $\hat{\mathbf{x}}_k$ describes the positions and behaviors of all agents.

The unique aspect of our implementation involves the correction step (2). The correction was implemented by first transforming to the *correspondence domain*, where the state distribution is mapped to a correspondence vector $\hat{\mathbf{c}} \in \mathbb{Z}^N$, with N equal to the number of simulated agents. Each entry of the correspondence vector is associated with an agent and indicates whether the agent is not associated with a sensor (in which case the entry is zero) or associated (in which case the entry is equal to the positive index of that sensor). The correspondence vector

978-1-7281-5092-5/19/$31.00 ©2019 IEEE

thus represents a hypothesis of possible associations between aggregate sensors and individual agents. Our algorithm mitigates combinatorial complexity by sampling a set of correspondence vectors from all possibilities. In all, the number M of sampled correspondence vectors is equal to the number of particles for each agent ($M = N = 50$ in this paper).

To transition from the prior (predicted) set of correspondence vectors to the posterior (corrected) set of correspondence vectors, a Metropolis-Hastings resampling step is introduced, a standard step in Markov Chain Monte Carlo (MCMC) estimation [23]. The uniqueness of our algorithm is that the Metropolis-Hasting step occurs in the correspondence domain [10]. The Metropolis-Hastings step involves sequentially selecting one of the sampled correspondence vectors $\hat{\mathbf{c}}$ and comparing it to a proposed alternative $\mathbf{c}'$. The alternative is created by a proposal process that must be reversible, but that is otherwise at the discretion of the algorithm designer; in our case, we selected a proposal process that created the alternative $\mathbf{c}'$ by randomly perturbing one entry of the original $\hat{\mathbf{c}}$ candidate, choosing the entry with uniform probability and "flipping" it from being associated with a sensor to unassociated (or vice versa). The original candidate and proposed alternative are then compared to compute an acceptance ratio a.

$$a = \min\left(1, \frac{p(\mathbf{y}_k|\mathbf{c}')p(\mathbf{c}')}{p(\mathbf{y}_k|\hat{\mathbf{c}})p(\hat{\mathbf{c}})}\right) \quad (3)$$

Note here that the proposal process is symmetric and thus the proposal distribution does not appear in (3).

Our Bayesian update step substitutes the proposed alternative $\mathbf{c}'$ for the original candidate $\hat{\mathbf{c}}$ with probability a. The process is then repeated with the goal of converging to a consistent posterior distribution. In MCMC algorithms, the number of substitutions is frequently chosen to be a fixed number $M+B$. In our case, M is the number of correspondence vectors and where B is an additional number of iterations that accounts for the transient or *burn-in* period. We set $B = 20$. Once convergence is achieved, the correspondence vectors are converted back to the *state domain* (location and behaviors) so that the prediction step (1) can be repeated.

III. Results

This section investigates individual-agent tracking performance for the Bayesian Estimation algorithm described above. Prior work has shown that tracking is quite good at the population level (when counting the total number of estimated agents at any one node) [10], but we have not previously investigated tracking at the individual-agent level (to see how well estimated agent states correspond to the associated member of the true population).

As a general statement, our estimator performed better tracking agents belonging to a group than tracking singleton agents. For instance, Fig. 2 below shows the distance error, or *DE*, for singleton agents (blue) is much higher than for grouped agents (green). The *DE* was evaluated as the distance of each particle's median location from the true location of the associated agent. For the purposes of visualization, a mean-absolute-deviation of *DE* was computed over the set of all particles associated with each agent. The mean-absolute-deviation score over all agents is shown in the figure. The shaded region represents the middle 40% of the agents, 20% above and below the median.

The network diameter for our location map is 14, meaning the shortest path between any two notes contains at most 14 edges. It is useful to normalize *DE* by network diameter; for instance, we might interpret "good" estimation as obtaining the answer within 20% of the network diameter. By this criterion, a "good" result would entail *DE* smaller than 2.8.

Applying this tracking-quality criterion to Fig. 2 reveals that the mean *DE* is high (above 2.8) at all times for singleton agents. This result confirms that the estimator cannot meaningfully distinguish among singleton agents. By contrast, Fig. 2 reveals that after time step 5 the mean *DE* is low (below 2.8) for agents belonging to groups. The early time steps represent an initial transient needed for the estimator to converge; after the transient is complete, "good" tracking ensues, meaning that grouped agents are distinguished based on their behaviors (e.g. to which rendezvous node they proceed). Even when confidence bounds (shaded regions) are considered to account for variability, it is clear that group performance is consistently better than singleton performance across the population of simulated agents.

It is interesting to note that *DE* does not decrease monotonically for the grouped agents, as seen by an increase in *DE* near time step 12. This increased DE is a result of mis-modeling. Whereas, in the ground-truth simulation, grouped agents transition deterministically from a rendezvous behavior to an evacuation behavior when all members of the group are reunited, the prediction step implements a simplified criterion that transitions randomly [10]. The consequence is some error in the timing of the behavior transition between the prediction and the ground truth. Once most group-member particles have transitioned (after time step 12), *DE* again decreases.

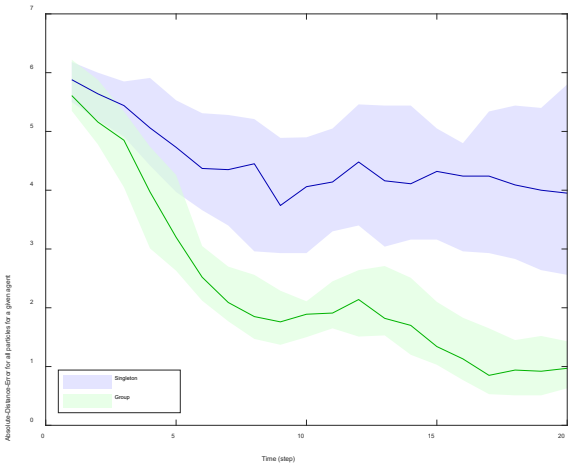

Fig. 1. Absolute Distance Error (DE) as a function of time. The interquartile range over all singleton agents (blue field) is significantly different than the interquartile range over all grouped agents (green field). Median values over singleton agents and grouped agents are shown as solid lines.

978-1-7281-5092-5/19/$31.00 ©2019 IEEE

To provide another view of the statistics, it is also useful to analyze the number of agents that meet the criterion for "good" tracking (with *DE* smaller than 20% of network diameter). These results are shown in Fig. 3. At first glance, the trends in this figure may appear opposite those in Fig. 2, but they are in fact consistent when the axis definitions are considered. In in Fig. 3, the vertical axis is the number of "good" particles for a given agent; this number should be high when *DE* is low. As expected, when average *DE* is low, then the number of particles below the *DE* cutoff (20% of network diameter) is high, as is the case for the group particles shown in the green regions in each figure. By contrast, when the average *DE* is high, then the number of particles that meet the DE cutoff is low, as is the case for the singleton particles shown in the blue regions of each figure.

It is clear from Fig. 3 that the number of "good" singleton particles (blue) converges quickly to approximately 25 particles (half of the total particle count $N = 50$). This result can be explained by noting the simulation includes two exits. Apparently, half of the singleton particles proceed to each exit. Once exited, the particle distribution does not significantly change. The exits are 8 edges apart, so if half the agents are at each exit, we would expect the average error to converge to a value of 4. (A DE of 4 is the weighted average with half the population at a *DE* of 0 and half at a *DE* of 8). Indeed, this hypothesis is consistent with the mean *DE* results, which converge to a DE of approximately 4, as shown in Fig. 2.

Tracking is better for the grouped agents because they proceed to known rendezvous points; this behavioral constraint provides important information leveraged by the estimator to improve tracking performance. The nearest exit to each rendezvous point is also known, so the estimator correctly predicts the exit node for nearly all grouped agents, as evidenced by the number of "good" group particles (green) in Fig. 3. The figure shows that, by the end of the simulation, nearly all particles (a mean of 49 out of 50) meet the criterion for "good" tracking.

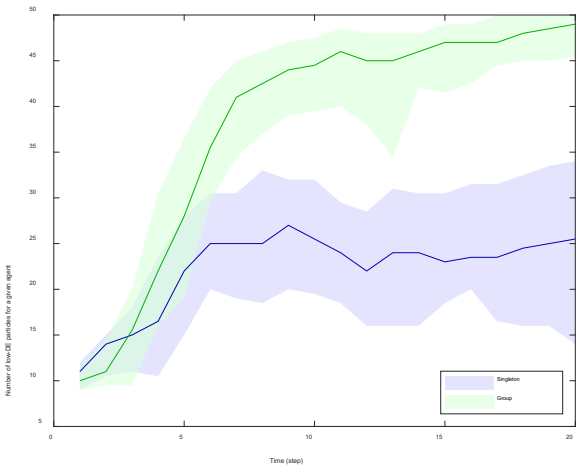

Fig. 2. Number of "good" (low DE) particles over time. The interquartile range of low-error particles over all singleton agents is shown in blue and over all grouped agents is shown in green. Median values are shown as solid lines.

Although they are well tracked, group agents still cannot easily be distinguished within their group (or even within the set of all groups meeting at a particular rendezvous point). The behavioral information available to the estimator is very much tied to knowledge about the destination (rendezvous point or exit) of each agent. In other words, all agents with the same destination are effectively equivalent from the point of view of the estimator as implemented.

IV. Discussion

Although the example here considers a relatively small "toy" simulation, the results strongly suggest that some identifying information can be extracted from aggregate sensors given an appropriately specific behavioral model. For instance, if home location, work location, and family relationships are known for a real individual, it is reasonable to assume that the individual's identity might be extracted by a real-world implementation of the Bayesian estimator described in this paper, even when the estimator fuses only aggregate sensor measurements. In the future, it will be informative to investigate the degree to which better directory information (e.g. census knowledge that could enable better inference of initial conditions) or more refined behavioral models might contribute to the identifiability of individual agents or of small subsets of agents. Also, it will be relevant to characterize how other types of sensor data (e.g. limited GPS position data transmitted by cell phones) might impact identifiability across the whole agent population.

V. Summary

This paper describes a Bayesian state estimator designed to fuse agent-based models with aggregate sensor data, such as sparse population counter measurements obtained by placing cameras throughout a physical space, such as a city. Results indicate that bulk population movements can be inferred. Inferences provide potential insight about population locations and behavior, even where no sensors are present. This information has significant potential utility for first responders addressing an unfolding disaster.

Perhaps surprisingly, our population tracker can also distinguish among agents, even when provided with sensor data that is inherently de-identified, such as aggregated population counts obtained over time. Individuals are identifiable, at least in simulation, through the fusion of integrated behavioral dynamics with the aggregate sensor data. Results from a simple simulation were presented that showed significant differences in tracking accuracy as a function of behavior. The capability to parse aggregate data to distinguish among individuals provides both potential benefits, to tailor disaster response to individual needs, but also some concern, as such a capability might compromise individual privacy.

Acknowledgements

Jason Rife gratefully acknowledges support from Air Force grant AFOSR FA9550-18-1-0465 and from National Science Foundation grant NSF CMMI-1903972, which partially supported this research. Samarth Swarup gratefully acknowledges support from DTRA CNIMS Contract

HDTRA1-17-0118 and NASA Grant # 17-HAQ17-0029, which partially funded this work. Nasim Uddin gratefully acknowledges partial support from USDOT STRIDE 2018-046 and NSF Grant # NSF-S&AS-1849264 for this work.

REFERENCES

[1] Buddemeier, B. R., Valentine, J. E., Millage, K. K., and L. D. Brandt, "National Capital Region: Key response planning factors for the aftermath of nuclear terrorism," *Technical Report LLNL-TR-512111*, Lawrence Livermore National Lab, November 2011.

[2] Federal Emergency Management Agency, "Planning guidance for response to a nuclear detonation," National Security Staff, Interagency Policy Subcoordination Committee for Preparedness and Response to Radiological and Nuclear Threats, 2010.

[3] Wein, L. M., Choi, Y., and S. Denuit, "Analyzing evacuation versus shelter-in-place strategies after a terrorist nuclear detonation," *Risk Analysis*, 30(9):1315–1327, 2010.

[4] Barrett, C. *et al.*, "Planning and response in the aftermath of a large crisis: An agent-based informatics framework," in R. Pasupathy, S.-H. Kim, A. Tolk, R. Hill, and M. E. Kuhl, editors, *Proceedings of the 2013 Winter Simulation Conference*, pages 1515–1526, Piscataway, NJ, USA, 2013. IEEE Press.

[5] Adiga, A., Marathe, M., Mortveit, H., Wu, S., and S. Swarup, "Modeling urban transportation in the aftermath of a nuclear disaster: The role of human behavioral responses," in *The Conference on Agent-Based Modeling in Transportation Planning and Operations*, Blacksburg, VA, Sep 30 - Oct 2 2013.

[6] Chanda, S. *et al.*, "Modeling the interactions between emergency communications and behavior in the aftermath of a disaster," in *The International Conference on Social Computing, BehavioralCultural Modeling, and Prediction (SBP)*, Washington DC, USA, April 2-5 2013.

[7] Franklin, G.F., Powell, J.D. and Emami-Naeini, A. *Feedback control of dynamic systems*. Prentice Hall Press, 2014.

[8] Simon, D. *Optimal state estimation: Kalman, H infinity, and nonlinear approaches*. John Wiley & Sons, 2006.

[9] Dunik, J., et al. "Random-point-based filters: Analysis and comparison in target tracking," *IEEE Transactions on Aerospace and Electronic Systems* vol. 51, no. 2, pp. 1403-1421, 2015.

[10] Lueck, J., Rife, J., Swarup, S., and N. Uddin, "Who goes there? Using an agent-based simulation for tracking population movement," *Proc. 2019 Winter Simulation Conference*, National Harbor, MD, 2019.

[11] Thrun, S., Burgard, W. and D. Fox. *Probabilistic robotics*. MIT press, 2005.

[12] Bar-Shalom, Y., and E. Tse, "Tracking in a cluttered environment with probabilistic data association," *Automatica*, Vol. 11, No. 5, pp. 451-460, 1975.

[13] Bar-Shalom, Y., Daum, F. and J. Huang, "The probabilistic data association filter," *IEEE Control Systems Magazine*, Vol. 29, No. 6, pp. 82-100, 2009.

[14] Farina, A. and S. Pardini, "Survey of radar data-processing techniques in air-traffic-control and surveillance systems," *IEE Proceedings F (Communications, Radar and Signal Processing)*, Vol. 127, No. 3, 1980.

[15] Bar-Shalom, Y., Kirubarajan, T. and X. Lin, "Probabilistic data association techniques for target tracking with applications to sonar, radar and EO sensors," *IEEE Aerospace and Electronic Systems Magazine*, Vol. 20, No. 8, pp.37-56, 2005.

[16] Khan, Z., Balch, T. and F. Dellaert, "An MCMC-Based Particle Filter for Tracking Multiple Interacting Targets," In *Computer Vision - ECCV 2004*, edited by T. Pajdla and J. Matas, 279–290, Springer Berlin Heidelberg, 2004.

[17] Smith, K., Gatica-Perez, D., and J-M. Odobez, "Using particles to track varying numbers of interacting people," *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, 2005.

[18] Kundu, P., *Fluid Mechanics*, Academic Press, Inc., pp. 48-49, 1990.

[19] Seo, T., Bayen, A.M., Kusakabe, T. and Y. Asakura, "Traffic state estimation on highway: A comprehensive survey," *Annual Reviews in Control*, vol. 43, pp.128-151, 2017.

[20] Bai, X. and Menon, P.K., "Optimal terminal area flow control using eulerian traffic flow model," in *AIAA Guidance, Navigation, and Control (GNC) Conference*, p. 5034, 2013.

[21] Tomastik, R., Lin, Y., and A. Banaszuk, "Video-based estimation of building occupancy during emergency egress," in proceedings of the *American Control Conference*, pp. 894-901, 2008.

[22] Bouarfa, S., *et al*., "Agent-based modeling and simulation of emergent behavior in air transportation," *Complex Adaptive Systems Modeling*, Vol. 1, No. 15, 2013.

[23] Khan, Z., Balch, T., and Dellaert, F. "MCMC-based particle filtering for tracking a variable number of interacting targets." *IEEE transactions on pattern analysis and machine intelligence*, vol. 27, no. 11, pp. 1805-1819, 2005.