Simultaneous Detection and Tracking of Multiple Objects in Noisy and Cluttered Environment using Maximum Likelihood Estimation Framework

Roman Ilin and Ross W. Deming*
Air Force Research Laboratory
Sensors Directorate
Hanscom AFB, MA
roman.ilin@hanscom.af.mil
*consultant for the Air Force Research Laboratory

***Abstract*-We discuss a versatile framework for multiple target detection and tracking based on maximum likelihood estimation with expectation maximization and a cognitive theory called dynamic logic. In this contribution extend the framework to detection of moving objects in video sequences. The paper presents the theory and an example of detection and tracking using a real world video sequence.**

I. Introduction

Target tracking refers to estimation of object's speed and position over time using one or multiple sensors. In the case of an electro-optical sensor, the tracking system analyzes a series of digital images or frames. The stages of tracking usually involve 1) target detection and track initiation, 2) track continuation, 3) track termination, and 4) identity declaration or classification. Each stage is complicated by the presence of multiple targets and other objects referred to as background clutter. Often information contained within a single frame is not sufficient to distinguish targets from clutter, making track initiation challenging. In such cases detection over a sequence of frame (multi-scan detection) can be employed. Multi-scan detection necessitates solving the data association problem, also known as the matching problem in the video tracking community [1], [2]. Solving the data association problem requires more computational resources as the number of possible combinations of data points grows exponentially.

Target motion often is modeled as linear, which will be accurate if the observation time interval is short enough. When multiple frames are superimposed on a single image the target track appears as a straight line. Thus, finding all the straight lines in the combined image is a good detection method. Successful methods for line detection include Hough transform [3] and matched filtering [4] among others. Unfortunately in a situation with high target and/or clutter density the number of candidate linear tracks becomes very large. Also, these methods cannot be easily extended to non-linear motion models.

As mentioned above, the number of possible tracks grows exponentially with the number of points on the image. The tracking algorithm has to determine which of the candidate tracks correspond to true tracks. Such determination can be made using probabilistic consideration, as in multiple hypotheses testing (MHT) approach [5]. Various heuristics are used to reduce the number of hypothesis that need to be tested. Such approach is suboptimal since valid hypothesis could be discarded in the process. An improvement over MHT is the introduction of probabilistic MHT (PMHT) [6], [7], [8], [9]. These approaches introduce probabilistic data association assigning the same point to multiple hypotheses. This soft assignment is augmented with the use of optimization techniques to find the optimal association probabilities. Proper choice of optimization technique is the key to avoiding combinatorial complexity of the problem.

The above discussion is related to tracking unresolved objects, which appear as dots on the image frames. Tracking resolvable objects involves simultaneous estimation of object's trajectory and appearance (size, shape, color, etc). These additional features often help detection and track initiation. For example, face detection or people detection algorithms allow identifying valid targets even on a single frame and thus avoiding multi-scan approach. However, in other real world scenarios, for example underwater video, when the types of targets are not known in advance, clutter density and noise are high, and the frame rate is low, the data association problem remains challenging even with resolvable targets. The additional features mentioned above still must be incorporated in the solution.

The essence of our approach is the simultaneous Maximum Likelihood Estimation of target track and appearance using Expectation Maximization (EM) approach to optimization [10]. Our methodology is based on a version of PMHT, independently developed in [11] based on the theory of cognition and perception called Dynamic Logic (DL). This approach has been successfully applied to multiple target tracking for unresolved targets using radar, optical and acoustic sensors[12]-[19] demonstrating robust performance and showing promise in overcoming the combinatorial complexity. We will refer to our methodology as EM/DL throughout this paper.

In order to apply the methodology to the task of tracking extended objects in video sequences we split tracking in 4 stages. During the *first* stage, we identify blobs on each image frame, using scale-space approach [20]. These blobs serve as input into the second stage of the algorithm. During the *second* stage the EM/DL algorithm identifies possible tracks formed by the blob centers. During the *third* stage, a different version of EM/DL identifies possible tracks based on the complete image information, using the results of the second stage as initial conditions. During the *fourth* and final stage of this process the less likely tracks identified in the thirds stage are discarded using a threshold. The third stage also includes pruning and merging of track models.

Section II of this paper will describe our methodology in detail. Section III contains an example of tracking with real video data. Section IV will discuss the results, the advantages of using our methodology, and the future research followed by conclusion in section V.

II. METHODOLOGY

The main components of EM/DL framework are the input data and the parametric track models. We denote the input pixel by $\mathbf{x}_n$, n=1..N. Each pixel is a vector with components giving its image coordinates, color, and timestamp. There are a total of N data points coming from multiple frames. We denote the models by M_h, h=1..H, where H is the total number of models. Each model h depends on parameters $\mathbf{S}_h$: $M_h=M_h(\mathbf{S}_h)$.

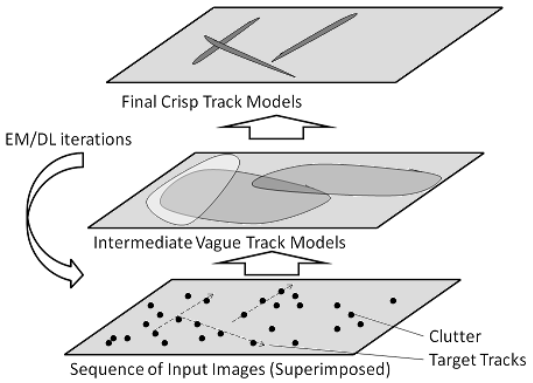

Figure 1. Schematic Illustration of the "vague-to-crisp" process emerging as result of EM/DL iterations. Multiple video frames (bottom) are superimposed. They contain the true target trajectories and many clutter points. The track models (middle) are initially vague covering large parts of the frame. The models become crisp at the end of the iterative process.

Models M_h are probabilistic, with conditional probability density function (PDF) of the data element $\mathbf{x}_n$ given the model M_h denoted by l(n|h). Note that this is a function of model parameters. The total conditional likelihood of the data is expressed as follows.

$$L(\boldsymbol{x}|\mathbf{M}) = \prod_{n=1}^{N}\left[\sum_{h=1}^{H} r_h l(n|h)\right] \qquad (1)$$

Here r_h is the a priori probability of encountering a model of class h. Maximization of (1) with respect to parameters results in the maximum likelihood parameter estimate. The maximization is subject to the following constraint

$$\sum_{h=1}^{H} r_h = 1 \qquad (2)$$

introducing an additional term with Lagrange multiplier into (1). Alternatively, these probabilities can be kept constant. The maximization of (1) is achieved by using the following iterative EM/DL procedure (3).

$$\begin{array}{ll}
1. & \boldsymbol{S}_h{}^{I} = \boldsymbol{S}_k{}^{0} \\
2. & f(h|n) = \dfrac{r_h l(n|h)}{\sum_{h_2=1}^{H} r_h l(n|h_2)}, given\ \boldsymbol{S}_h{}^{I} \\
3. & \begin{cases} \displaystyle\sum_{n=1}^{N} f(h|n)\frac{\partial \log l(n|h)}{\partial \boldsymbol{S_h}} = 0 \ \longrightarrow\ \boldsymbol{S}_h{}^{I+1} \\ r_h = \dfrac{\sum_{n=1}^{N} f(h|n)}{N}\ \ (could\ also\ be\ given\ a\ priori) \end{cases} \\
4. & \boldsymbol{S}_h{}^{I} = \boldsymbol{S}_h{}^{I+1} \\
5. & Adjust\ DL\ parameters\{\ \boldsymbol{S_h}^{\boldsymbol{I+1}}\ \}_{DL}
\end{array} \qquad (3)$$

This procedure is derived in [11] by considering a special form taken by the derivatives of log L(**x**|**M**) with respect to model parameters $\mathbf{S}_h$ and introducing association weights f(h|n). Steps 1-4 of this procedure can also be derived using the general EM principles and interpreting f(h|n) as missing data [10]. The essence of (3) is the maximization of L(**x**|**M**) with respect to the association weights given the current estimates of the model parameters followed by maximization of L(**x**|**M**) with respect to the parameters. It is known that such iterative procedure always results is increase of L(**x**|**M**) . It is also known that the rate of convergence of EM procedure is constant in the neighborhood of the maximum [10]. This means that if the parameters are initialized close enough to their true values the procedure will quickly converge. Improper initialization, on the other hand, can result in slow convergence or convergence to a local maximum. This is where the principles outlined in [11] provide a solution. The Dynamic Logic theory suggests that as long as a gradual transition from vague to crisp data association is maintained the procedure will converge quickly and avoid local maxima. This transition is enforced by proper initialization of the model parameters and scheduled annealing of some of the model parameters, referred to as "DL parameters" in step 5 of (3).

Figure 1 illustrates this idea. The bottom image contains the data from multiple frames with true tracks marked by arrows. The track models are initialized with high degree of uncertainty about the location of the tracks. These initial vague track models are shown in the middle image. As the EM/DL procedure converges, the uncertainty about track locations decreases making the models crisper until they finally converge to the true tracks. Such convergence has been observed in many applications including this contribution, even though more theoretical work needs to be done in order to fully understand the DL performance.

Practical application of EM/DL algorithm consists of specifying the exact form of l(n|h) for each model type. In the general case, which covers resolvable objects, the track model consists of three components: object model, feature model and motion model. This is expressed as the following product of PDF's.

$$l(n|h) = l_{obj}(n|h)\, l_{feat}(n|h) l_{mot}(n|h) \quad (4)$$

The specific forms of PDF's depend on application. In our case we use two different sets of PDF's for stage 2 and 3 of processing. We will describe each stage of processing in the following subsections.

Stage 1: Blob Detection

At this stage blobs are detected on each input frame using scale space approach [20]. Each image is convolved with a series of Laplacian of Gaussian kernels with predefined sequence of standard deviations (scales) to form a series of scale-space images. For each scale, the blobs result in maxima of the scale-space image. These maxima are extracted and their mean intensities, locations and scales become the output of the first stage. We specify the maximum number of blobs to return from this stage to avoid insignificant maxima. Thus, the output of stage 1 is a set $\{\mathbf{cx}_b, rad_b, y_b, t_b, b=1..B\}$ where B is the total number of blobs extracted from the frames, $\mathbf{cx}_b$ is vector of coordinates of the blob center, measured in pixels, rad_b, is the radius of the blob corresponding to the scale, y_b is the gray scale intensity of the original image pixel corresponding to the blob center, and t_b is the timestamp of the blob identifying which frame it came from.

Stage 2: Tracking blob centers

Stage 2 consists of applying the iterations (3) to the data from stage 1. We specify the following PDF's for the track model. The feature model describes the gray scale intensity of pixels with Gaussian density

$$l_{feat}(n|h = target) = \frac{e^{-0.5(y-Y_h)^2/{\sigma_h}^2}}{\sqrt{2\pi{\sigma_h}^2}} \quad (5)$$

with unknown parameters σ_h and Y_h. The motion model is given by the following PDF

$$l_{mot}(n = target|h) = \frac{e^{-0.5(c\boldsymbol{x}_n - X_{hn})^T {\boldsymbol{CX}_h}^{-1}(c\boldsymbol{x}_n - X_{hn})}}{\sqrt{(2\pi)^2|\boldsymbol{CX}_h|}} \quad (6)$$

where

$$\boldsymbol{X}_{hn} = \boldsymbol{X0}_h - \boldsymbol{V}_h t_n \quad (7).$$

Thus the unknown parameters of the motion model include the initial position and velocity of the linear track $\boldsymbol{X0}_h$ and $\boldsymbol{V}_h$ and the variance $\boldsymbol{CX}_h$ corresponding to the uncertainty of the track location. The object model is not used since we track point targets and therefore $l_{obj}(n|h = target) = 1$.

The model for background clutter is an essential part of the algorithm, since every pixel must correspond to one of the models in order for EM/DL process to successfully converge. We can specify the clutter model using the three components (4). As in the case of target, $l_{obj}(n|h = clutter) = 1$. The feature model is modeled with Gaussian density.

$$l_{feat}(n|h = clutter) = \frac{e^{-0.5(y-Y_h)^2/{\sigma_h}^2}}{\sqrt{2\pi{\sigma_h}^2}} \quad (8)$$

The motion model is modeled with uniform distribution

$$l_{mot}(n|h = clutter) = \frac{1}{A} \quad (9)$$

where A is the area of the image in pixels. This means that the clutter pixel can appear in any position on the frame and it does not move.

Finally, we use $\boldsymbol{CX}_h$ as the DL parameter to enforce the vague-to-crisp transition. This parameter is set as follows.

$$\boldsymbol{CX}_h(l) = \boldsymbol{CX}_e + (\boldsymbol{CX}_e - \boldsymbol{CX}_s)e^{-\tau(l+1)} \quad (10)$$

where $\boldsymbol{CX}_e$, $\boldsymbol{CX}_s$, and τ are parameters selected empirically.

Equations (5-10) together with (3) give the complete specification of the stage 2 algorithm. In order to implement the algorithm all the derivatives necessary in step 3 of (3) need to be derived along with ways to solve the step 3 equations for parameter values.

Stage 3: Tracking extended objects

Stage 3 consists of another application of (3), this time using the complete image data. The following PDF's are used for the track model. The feature PDF is exactly the same as in stage 2 given by (5). The motion PDF is given as follows

$$l_{mot}(n|h = target) = \frac{e^{-0.5(\boldsymbol{xc}_{hn} - \boldsymbol{X}_{hn})^T {\boldsymbol{CX}_h}^{-1}(\boldsymbol{xc}_{hn} - \boldsymbol{X}_{hn})}}{\sqrt{(2\pi)^2|\boldsymbol{CX}_h|}} \quad (11)$$

Here the parameters of the PDF are $\boldsymbol{CX}_h$ and $\boldsymbol{X}_{hn}$ with the latter defined by (7). The difference is that the random variable of this PDF is $\boldsymbol{xc}_{hn}$ - the coordinates of the object's center of motion. This variable by itself is an unknown parameter related to the image pixel data $\boldsymbol{x}_n$ through the object model as follows.

$$l_{obj}(n|h = target) = \frac{e^{-0.5(x_n - xc_{hn})^T {\boldsymbol{CO}_h}^{-1}(x_n - xc_{hn})}}{\sqrt{(2\pi)^2|\boldsymbol{CO}_h|}} \quad (12)$$

The object model describes the shape of the object. We model the targets with Gaussian density using the fact that most of the density is concentrated within two standard deviations from the mean. Therefore $\boldsymbol{CO}_h$ is used as a parameter describing the size and shape of the target. The shape is specified with respect to the object's motion center $\boldsymbol{xc}_{hn}$.

The background model is the same as in stage 2. Finally the DL parameter $\boldsymbol{CX}_h$ is set to a constant value and is not changed during the iterations. There is no need to use this parameter since the initial conditions are set using the results

of stage 2. We use the final values of $\boldsymbol{X0}_h$ and $\boldsymbol{V}_h$ from stage 2 and initial values for the motion model at this stage. The other parameters: feature and object centers and object sizes are estimated during stage 3.

Stage 3 also includes merging and pruning of models.

Stage 4: Detection

At this stage the parameters of all models have been adjusted to the values that provide the best fit between the data and the models. Since the models are rather simple we need to use additional criteria to identify those models that converged to valid targets. Since we are looking for elliptically shaped objects, we discard those models that are not adequately "filled" by the data.

At the end of Stage 3 the association weights f(h|n) represent the assignment of pixels to different models. This is a soft assignment since each pixel can have several non-zero association weights linking it to different models. In order to obtain a hard assignment we take the model corresponding to the maximum weight for each pixel. Formally we define an indicator variable $\alpha_n \in [1\,..H]$ such that

$$\alpha_n = \arg\max_{h=1..H} f(h|n)\ , \qquad n = 1..N \quad (13).$$

The number of pixels assigned to model h is then given as follows.

$$N_h = \sum_{n=1}^{N} \delta_{\alpha_n h}\ , h = 1..H \quad (14)$$

Here δ is the Kronecker delta. We define the fill factor of model h as the ratio of the number of pixels assigned to it to the area of the object model proportional to the product of the eigenvalues of the model's covariance matrix $\boldsymbol{CO}_h$.

$$F_h = \frac{N_h}{\max(eig\ \boldsymbol{CO}_h)\min\,(eig\ \boldsymbol{CO}_h)}\ , h = 1..H \quad (15)$$

This fill factor is compared to a predefined threshold to make the detection decision.

III. Example of Tracking

In this section we apply the steps described in the previous section to a sequence of video frames taken by a digital camera at the local aquarium. The sequence consists of 7 frames with the size of 320 by 240 pixels. The frame rate was 15 frames per second. Even though the camera recorded colored images we converted them to gray scale for these experiments. One of the frames is shown in Fig. 2. In addition to moving fish the image contains many plants and rocks, which can be mistaken for a fish especially when gray scale image is considered. The camera is hand held resulting in jitter and induced motion of the objects from frame to frame. The goal of applying our algorithm was to detect moving fish.

The results of the first stage of processing are shown in Fig. 3 for one o f the frames. We limited the number of blobs to 60 per frame.

The second stage of processing is illustrated in Fig 4. The black dots correspond to blob centers. Each image in Fig. 4 contains the blob centers from all 7 frames superimposed. The white cloud-looking parts of the image represent the track models. We used 24 track models and one clutter model. The size of each cloud corresponds to the track position uncertainty $\boldsymbol{CX}_h$. During the first iterations the uncertainty is high and all the models cover most of the image and overlap each other. As the uncertainty decreases the models focus on different parts of the image identifying possible tracks. This process requires only 20 iterations to complete. The resulting tracks are now used for the next stage of processing shown in Fig 5. We show only 3 frames out of 7. The algorithm initialized the object models using blob size information from stage 2. The iterations of EM/DL algorithm adjust each model to match the object shape and trajectory. As in the previous stage, we use only 20 iterations of the algorithm.

Figure 2.One frame of the fish tank sequence.

The models converge on fast moving fish and some rocks or plants. During this stage we also merge models converging to the same target. This is done by checking the Euclidean distance between the track models and merging the models that are close to each other, as determined using a predefined threshold.

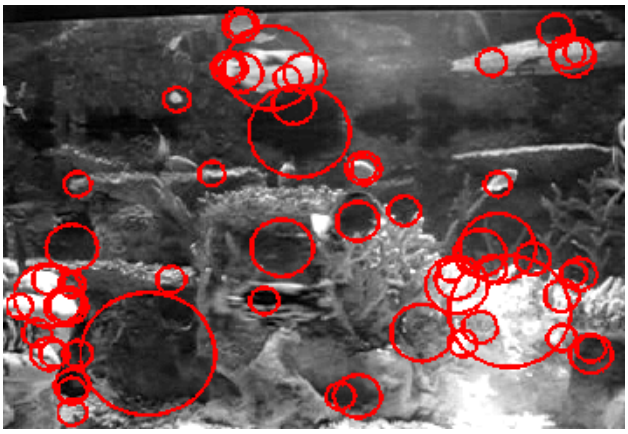

Figure 3. Blobs detected on the frame shown in Fig. 2.

The final stage of processing includes the computation of fill factors (15) for each model and making the detection decision. Fig. 6 shows the resulting models. Two models converged to fast moving fish. Four other models converged to plants or fragments of rocks with high fill factor.

IV. Discussion

The example in section 3 demonstrated the stages of EM/DL algorithm applied to a video sequence. As we mentioned in the introduction, multiple target tracking is difficult due to combinatorial complexity arising from the association problem. Accordingly, the main advantage of our algorithm appears in stage 2 where the Dynamic Logic process allows quick identification of linear tracks in the sequence of images. We would like to list some other advantages of using this type of approach.

The essence of EM/DL approach is in combining existing knowledge about potential targets with adaptive mechanism capable of finding a quick match between the models representing this knowledge and the data coming from the sensor. This idea is deeply rooted in psychology and neurophysiology. From the times of Helmholtz, it was understood that the eye itself is not a very precise instrument. Helmholtz suggested that object recognition and tracking is performed using previous experience stored in the brain [21]. This means that in addition to the bottom-up signals coming from the retina the brain uses top-down signals to guide the process of recognition. The use of prior experience in biological vision is actively researched theoretically and empirically [22], [23], [24], [25]. In the case of our algorithm, the improvement of knowledge translates into more precise parametric models. The EM/DL algorithm allows quick replacement of model types without changing other parts of the computer code.

In our example the algorithm detected only two fast moving fish. Looking at the image we can see that there are several more fish targets left undetected. The reason for this is the use of linear motion model. The three small fish in our sequence move very slowly. In addition, the hand held camera adds random motion resulting in highly nonlinear sequence of blob centers. This problem can be solved with the introduction of other motion model types.

The same solution applies to more complex target types. Any number of additional features such as more complex shape, color, and texture can be easily incorporated into the feature model without changing the other parts of properly designed computer code.

The algorithm also is easily amenable to parallel and distributed implementations important for many critical applications. The evaluation of association weights and model parameters is independent for each model and therefore can be done in parallel. Real time applications can be programmed on specialized hardware such as Field Programmable Gate Arrays (FPGA).

The algorithm can be used for processing data from multiple sensors with low communication bandwidth. This is achieved by modifying the algorithm to maintain separate set of model parameters for each sensor for local processing and combining them at a central location without sending the entire data set. Work on such applications is currently under way.

Finally, this framework is not exclusive to video processing. The same methodology has been successfully used for radar and sonar data, and even symbolic processing [26]. The strength of this approach is in the universal applicability of the core algorithm, making it a strong candidate for the next generation cognitively based sensor networks [27].

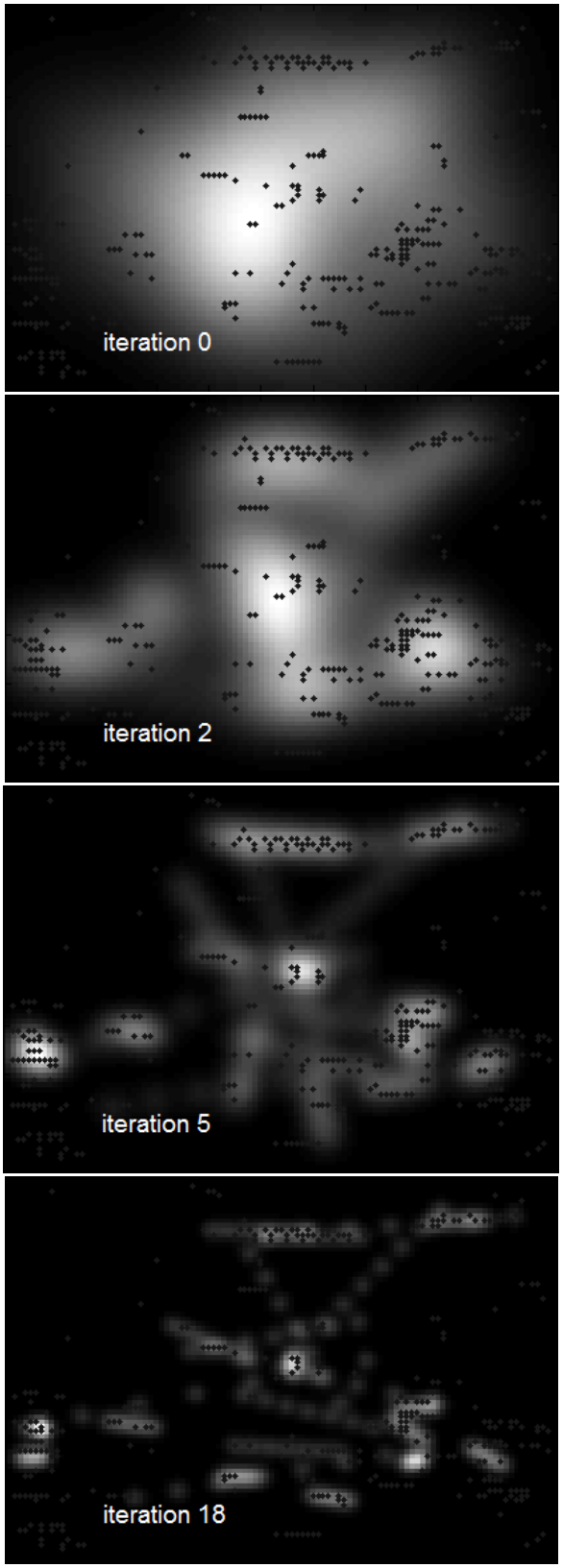

Figure 4.EM/DL iterations with blob centers as inputs. The black dots are the centers of the blobs detected on each frame similar to Fig 3. The white "clouds" are the PDF's of models. In the beginning of the process the models cover large portions of the input frames and at the end they converge on the most likely linear tracks. Note that the target shape information is not used at this stage of detection and tracking.

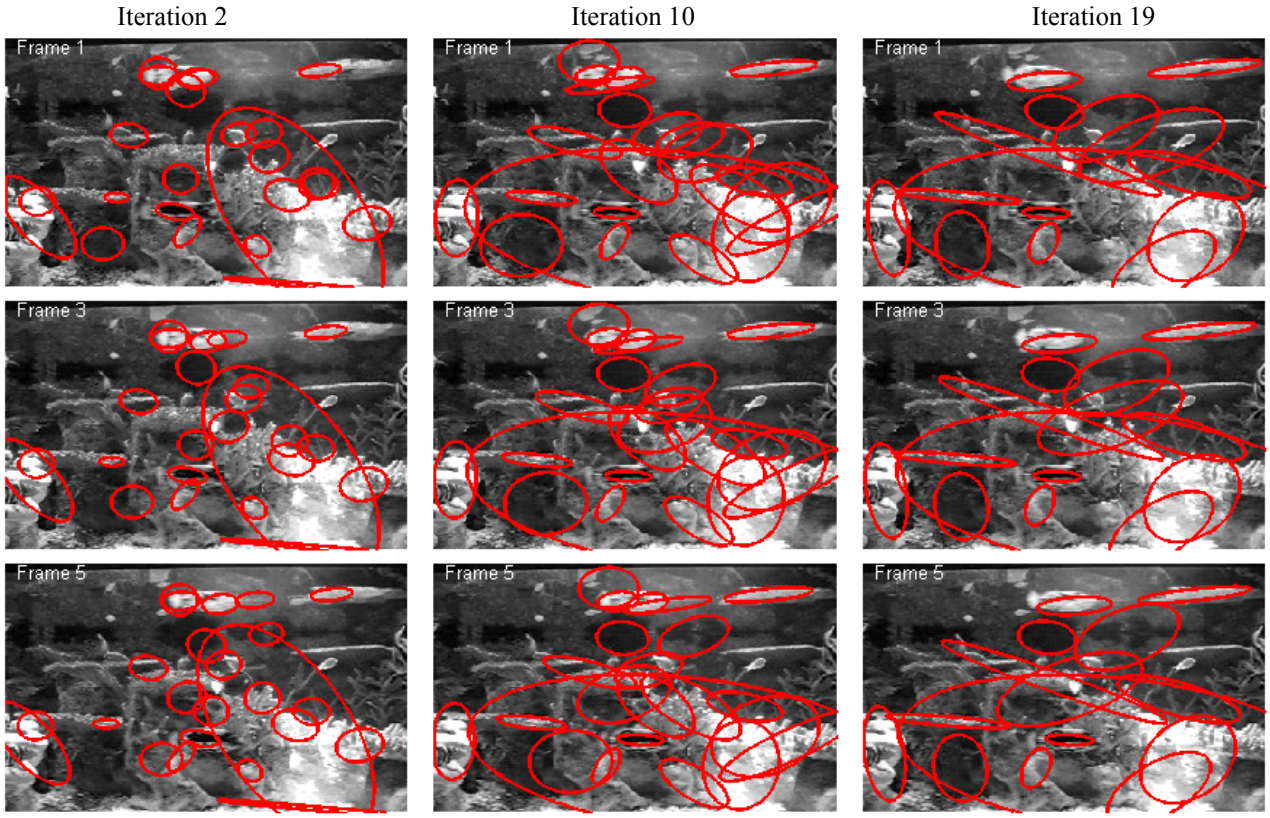

Figure 5. EM/DL iterations using entire image frames as inputs and the tracks detected in Fig. 4 as initial conditions. In addition to the brightness and motion information, the track models now include the shape information. The shape models are shown as the red ellipses. During the iterations the ellipses adapt to the objects in the images and the initial position and velocity of each track adapt to better correspond to the target truth. The figures also illustrate how models converging on the same track merge at the end of the process. This can be seen by looking at the large fish in the top-middle part of the frame.

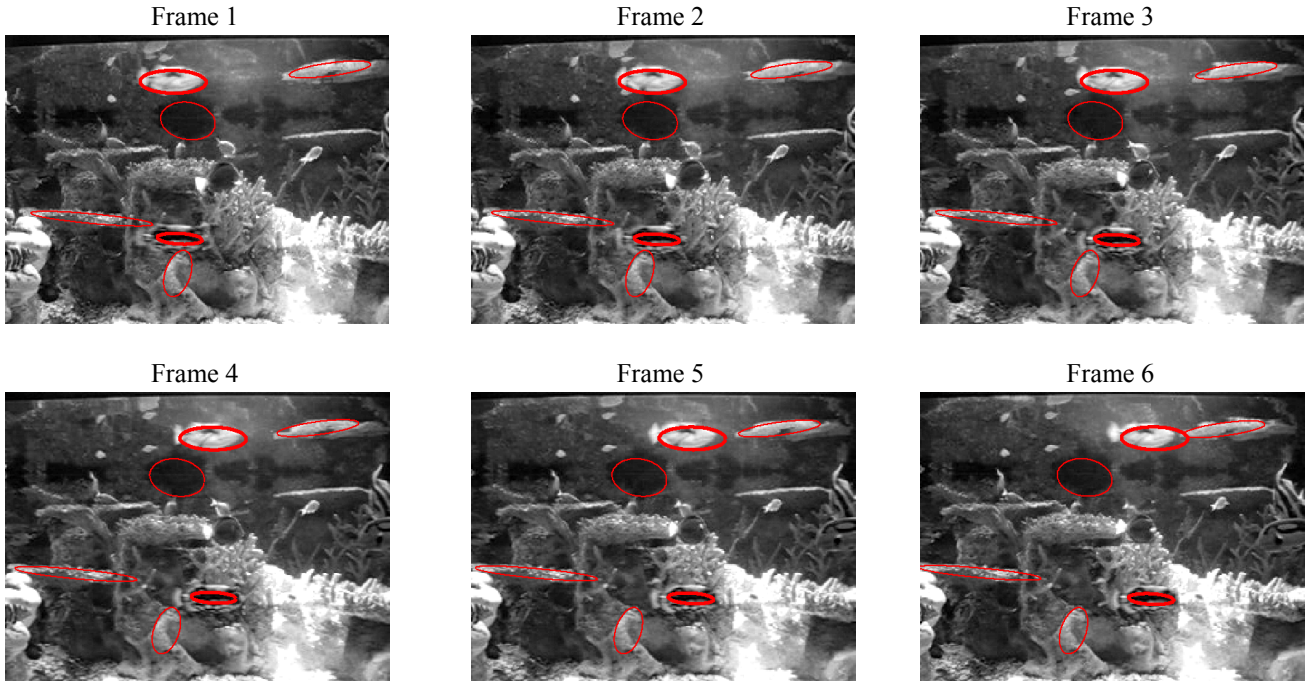

Figure 6. Blobs detected on the frame shown in Fig. 2. Tracks remaining after a detection threshold as described in the text. Two tracks (thick ellipses) correspond to fast moving fish. Four other models (thin ellipses) correspond to elliptically shaped objects slow moving due to the motion of the hand held camera. The other slow moving fish were not detected due to highly non-linear trajectories.

V. Conclusion

This work demonstrated an application of probabilistic ML framework called Dynamic Logic to video tracking of multiple targets. We described the challenges related to this problem and presented our methodology and algorithm. Even though DL has many successful applications, it has not been applied to image processing in the past. The goal of this work was to demonstrate the applicability of this approach to video tracking, describe its advantages as a generic cognitive based algorithm, and outline the directions of future research.

Acknowledgment

The authors would like to thank the United States Air Force for continuing support of this research. This work was supported in part by AFOSR under the Laboratory Task 05SN02COR, PM Dr. Jon Sjogren, by Laboratory Task 08RY02COR, PM Dr. Doug Cochran and by NRC fellowship.

References

[1] Y. Bar-Shalom and T. Fortmann, *Tracking and Data Association*. San Diego: Academic Press, Inc, 1988.

[2] E. Trucco and K. Plakas, "Video Tracking: A Concise Survey," *IEEE Journal of Oceanic Engineering*, vol. 31, no. 2, pp. 520-529, Apr. 2006.

[3] B. Carlson, E. Evans, and S. Wilson, "Search radar detection and track with the Hough transform. I. system concept," *Aerospace and Electronic Systems, IEEE Transactions on*, vol. 30, no. 1, Jan. 1994.

[4] B. Porat and B. Friedlander, "A frequency domain algorithm for multiframe detection and estimation of dim targets," *IEEE Trans. Pattern Anal. Machine Intell*, vol. 12, no. 4, 1990.

[5] D. Reid, "An Algorithm for Tracking Multiple Targets," *Automatic Control, IEEE Transactions on*, vol. AC-24, no. 6, Dec. 1979.

[6] R. L. Streit and T. E. Luginbuhl, "Maximum likelihood method for probabilistic multi-hypothesis tracking," in *Proceedings of SPIE International Symposium, Signal and Data Processing of Small Targets*, Orlando, FL, Apr. 5-7,1994.

[7] P. Willett, Y. Ruan, and R. Streit, "PMHT: Problems and some solutions," *IEEE Transactions on Aerospace and Electronics Systems*, vol. 38 , pp. 738-754, 2002.

[8] Y. Ruan and P. Willett, "Multiple model PMHT and its application to the second benchmark radar tracking problem," *IEEE Transactions on Aerospace and Electronics Systems*, vol. 40 , pp. 1337-1347, 2004.

[9] D. Avitzour, "A maximum likelihood approach to data association," *IEEE Transactions on Aerospace and Electronics Systems*, vol. 28, pp. 560-565, 1992.

[10] G. J. McLachlan and T. Krishnan, *The EM algorithm and extentions*. John Wiley & Sons, Inc, 1997.

[11] Perlovsky, *Neural Networks and Intellect*. Oxford, 2001.

[12] R. Deming, "Automatic Buried Mine Detection Using the Maximum Likelihood Adaptive Neural System (MLANS)," in *Proc. of the 1998 IEEE ISIC/CIRA/ISAS Joint Conference*, Gaithersburg, MD, Sept. 14-17, 1998.

[13] R. Deming and L. Perlovsky, "Concurrent multi-target localization, data association, and navigation for a swarm of flying sensors," *Information Fusion*, vol. 8, no. 3, pp. 316-330, 2007.

[14] R. Deming, J. Schindler, and L. Perlovsky, "Concurrent Tracking and Detection of Slowly Moving Targets using Dynamic Logic," in *2007 IEEE Int'l Conf. on Integration of Knowledge Intensive Multi-Agent Systems: Modeling, Evolution, and Engineering (KIMAS 2007)*, Waltham, MA, April 30 – May 3, 2007.

[15] R. Deming, J. Schindler, and L. Perlovsky, "Multitarget/Multisensor Tracking using only Range and Doppler Measurements," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 45, no. 2, Apr. 2009.

[16] R. Deming, J. Schindler, and L. Perlovsky, "Track-Before-Detect of Multiple Slowly Moving Targets," in *IEEE Radar Conference 2007*, Waltham, MA, April 17-20, 2007.

[17] L. I. Perlovsky and M. M. McManus, "Maximum Likelihood Neural Networks for Sensor Fusion and Adaptive Classification," *Neural Networks*, vol. 4, no. 1, pp. 89-102, 1991.

[18] L. I. Perlovsky and R. W. Deming, "Neural Networks for Improved Tracking," *Neural Networks, IEEE Transactions on* , vol. 18, no. 6, pp. 1854-1857, Nov. 2007.

[19] A. Schatzberg, A. Devaney, and R. Deming, "Maximum Likelihood Estimation of Target Location in Acoustic and Electromagnetic Imaging," in *Expanded Abstracts from the 65th meeting of the Society of Exploration Geophysicists*, Houston, TX, Oct. 1995.

[20] T. Lindeberg, "Feature detection with automatic scale selection," *International Journal of Computer Vision*, vol. 30, no. 2, pp. 77--116, 1998.

[21] Helmholtz, *Helmholtz's Treatise on Physiological Optics*. 1866/1924: The Optical Society of America.

[22] W. J. Freeman, "How brains make chaos in order to make sense of the world," *Behavioral and Brain Sciences* , vol. 10, pp. 161-195, 1987.

[23] S. Grossberg, *Neural Networks and Natural Intelligence*. Cambridge, MA: MIT Press, 1988.

[24] C. Q. Howe, R. B. Lotto, and D. Pruves, "Comparison of Bayesian and empirical ranking approaches to visual perception," *Journal of Theoretical Biology*, vol. 241, pp. 866-875, 2006.

[25] M. Bar, et al., "Top-down facilitation of visual recognition," *Proceedings of the National Academy of Sciences*, vol. 103, pp. 449-54, 2006.

[26] R. Ilin and L. I. Perlovsky, "Cognitively Inspired neural network for Recognition of Situations," *International Journal of Natural Computing Research*, 2010.

[27] S. Haykin, "Cognitive radar: a way of the future," *IEEE Signal Processing Magazine*, 2006.