A Hierarchical Behavior Analysis Approach for Automated Trainee Performance Evaluation in Training Ranges

Saad Khan, Hui Cheng, and Rakesh Kumar

SRI International, Princeton, USA
{saad.khan,hui.cheng,rakesh.kumar}@sri.com

Abstract. In this paper we present a closed loop mixed reality training system that provides automatic assessment of trainee performance during kinetic military exercises. At the core of our system is a hierarchical behavior analysis approach that integrates a number of data sensor modalities including Audio/Video, RFID and IMUs to automatically capture trainee actions in a comprehensive manner. Our behavior analysis and performance evaluation framework uses a finite state machine (FSM) model in which trainee behaviors are the states of the training scenario and the transitions of states are caused by stimuli that we refer to as trigger events. The goal of behavior analysis is to estimate the states of the trainees with respect to the training scenario and quantify trainee performance. To robustly detect each state, we build classifiers for each behavioral state and trigger event. At a given time, based on the state estimation, a set of related classifiers are activated for detecting trigger events and states that can be transitioned to and from the current states. The overall structure of the FSM and trigger events is determined by a Training Ontology that is specific to the training scenario.

1 Introduction

Infantry training, from basic training at home stations to joint exercises prior to deployment, can become more effective through automated behavior analysis and performance evaluations. In this paper, we present an automated behavior analysis and performance evaluation computational framework for a wide range of training objectives.

We model trainee behavior (individually and in teams) as states, and the causes of state transitions as trigger-events. Each state has a set of performance metrics. The overall goals of the training exercise are captured as hierarchical Finite State Machines (FSM) with associated performance metrics. Our behavior analysis module uses sensor data as observations to estimate the states that the trainees are in. The performance evaluation module computes the performance metrics given the estimated states of the trainees. Trigger events that result in transition from one state to another are detected using a Histograms of Oriented Occurrence (HO2) algorithm for

D.D. Schmorrow and C.M. Fidopiastis (Eds.): AC/HCII 2013, LNAI 8027, pp. 60–69, 2013.

individual and group activity recognition that captures the interactions of multiple players in one feature vector.

The system uses a suite of multi-modal sensors to capture training exercise (see figure 1). Each trainee's location, weapon and head orientations are computed using a combination of GPS or RFID, inertia navigation sensors (INS) data and video analysis. Gunshots are captured through trigger sensors and laser shot detection system. Videos and detected events are overlaid on a 3D-model of the training site for enhanced AAR and situational awareness experiences. Additionally, our AAR allows searching and browsing of training events and the computation of statistics. Our system estimates behaviors and corresponding performance metrics in real-time, and ingests both into a database. Experimental results from out prototype training system have shown improved training efficiency and effectiveness as a result of the system.

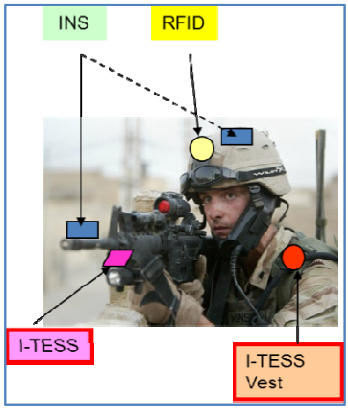


Fig. 1. Multi-modal sensor suite used to instrument trainees

2 Automated Behavior Analysis and Event Detection

Our training domain (military exercises) circumscribes the space of observable behaviors to a controllable list. Taking advantage of this fact we are able to develop a behavior analysis framework that uses a finite state machine (FSM) model where participants' behavior are the *states* and the transitions of states are caused by stimuli that we refer to as *trigger events*. The goal of behavior analysis is to estimate the states of the participants and the states that the participants should be in at any given time. The former are used for exercise and scenario control and the later are used for performance evaluation. To robustly detect each state, we build classifiers for not only for each state, but also for each trigger event. At a given time, based on the state estimation, a set of related classifiers are activated for detecting trigger events and states that can be transitioned to and from the current states.

We model a training exercise as a finite state machine (FSM). A FSM is a quintuple $(\Sigma, S, s_0, \delta, F)$, where:

- Σ is the input alphabet (a finite and non-empty).
- S is a finite, non-empty set of states.
- s_0 is an initial state, an element of S.
- δ is the state-transition function that returns a set of transition probabilities: $\delta : S \times \Sigma \rightarrow P(S)$.
- F is the set of final states, a subset of S.

For training,

- Σ is the set of stimuli or trigger events
- *S* is the set of possible behaviors, i.e. states of the participants.
- s_0 is an initial state.
- δ is the reaction to a stimulus. δ contains both the correct reactions to stimuli defined in a TTP and incorrect reactions that need to avoid.
- *F* is the end state of a training exercise.

For a training system, states *S* can only be perceived through sensor observations, *O*. Then, behavior analysis is to estimate states $S=\{s_0,s_1,..,s_n\}$ given sensor observation $O=\{o_0,o_1,..,o_n\}$. In our system, the sensor inputs include positions of all participants, their head, body and gun poses and shot/hit data (figure 1). However the definition of state *S* and transition trigger events depends on the Training Ontology discussed next.

2.1 Training Ontology

The training ontology captures knowledge related to a set of training objectives including TTP (Techniques, Tactics and Procedures), training scenarios and performance metrics. This is a machine understandable graphical-representation of the TTP that includes comprehensive data on scenario context, parameters for behavior recognition, and expected performance evaluation thresholds. Our training taxonomy is divided into two sub-hierarchies – a set of concepts representing states (nouns) and a set representing trigger events (verbs). Using Protégé [Noy, 2001], we assign a node to each state, along with the corresponding definition. Similarly, we assign a node to each trigger event and its definition. All states and trigger events form the taxonomy in our training ontology. For each state, we also store associated attributes including classifier and the performance metrics for the state. For each state and a given trigger event, the ontology also captures all states that it can transition to. Figure 2 illustrates an example exercise model that includes speech and gestures so that they can be assessed in the same overall framework.

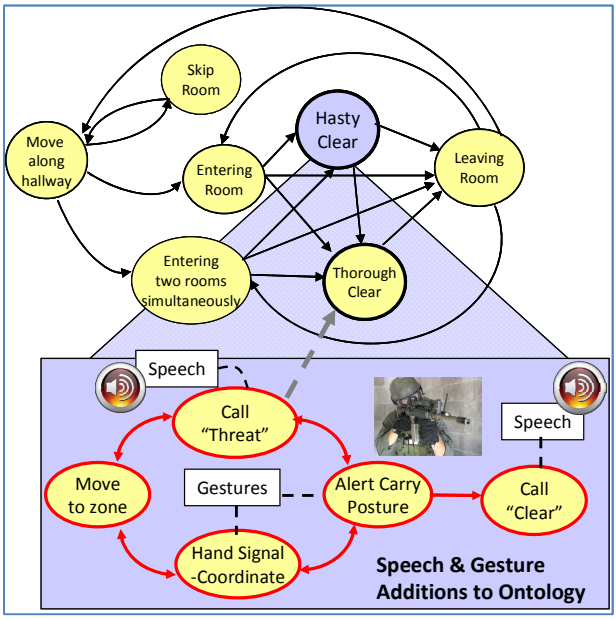


Fig. 2. Example Ontology of Exercise Room Clearing

2.2 Hierarchical Behavior Analysis

The training ontology helps us define the FSM that represents only the top layer of our hierarchical behavior analysis module. As illustrated in figure 3 at the lowest level is the Action Detection module that classifies atomic actions performed by the participants. These atomic actions span a wide array of low-level trainee behaviors like "walking", "group formation", "weapon sector scanning", "weapon fire" etc. In most cases classifiers for these atomic actions are trained on static features extracted directly from the raw sensor data. For instance to detect "group formations" the track locations of the trainees are used to match against a shape template pertaining to a "diamond" or "wedge" formation. In the middle layer, we generate Trigger Events which are mid-level abstractions of trainee behavior that result in a meaningful transition from one state in the scenario to another. These trigger events typically represent a dynamic activity that require features to be extracted over a window of time frames. Figure 3 illustrates some examples of these including "Cordon Formation", "Crossed Danger Zone" etc.

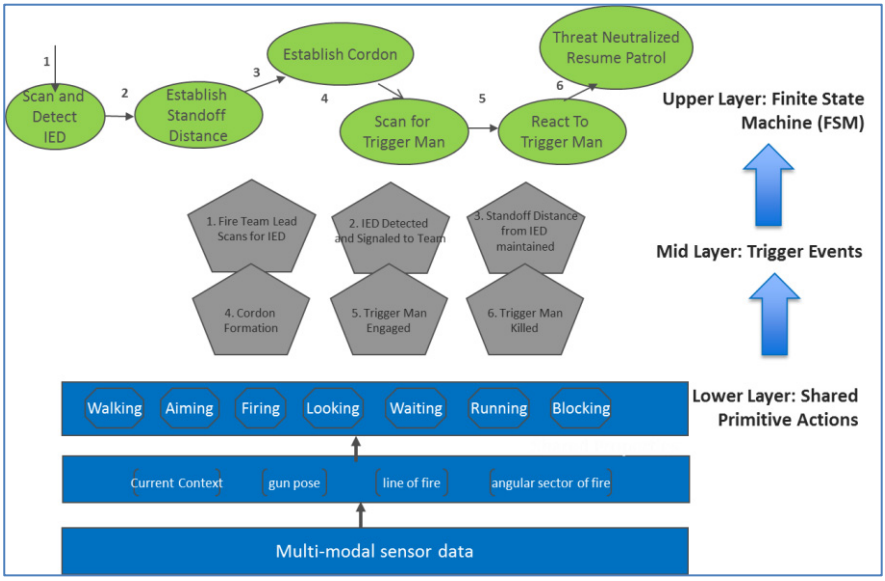


Fig. 3. Hierarchical framework for behavior analysis

Adaptive space-time aggregated features Histogram of Oriented Occurrences (HO2) are computed and trained with SVM to classify atomic actions and trigger events. In its most generalized form, space-time context is the histogram of occurrences of entity classes of interest over a partition of a spatial-temporal volume with respect to a reference entity or a reference location. Existing activity or event exploitation approaches represent these events using features that only measure pair wise relationships between entities at a time, such as relative distance and relative speed. Due to the limitations of the pair wise entity relationship descriptors, this class of

events is mainly defined and recognized using rule-based approach. HO2 captures the interactions of all entities of interests in terms of configurations over space and time through a histogramming process. Using this new space-time context representation, our activity exploitation approach captures both environmental context and spatial-temporal characteristics of the entities in a unified framework. Using HO2, we have been able to detect multi-agent events such as VIP arriving or depicturing with security details.

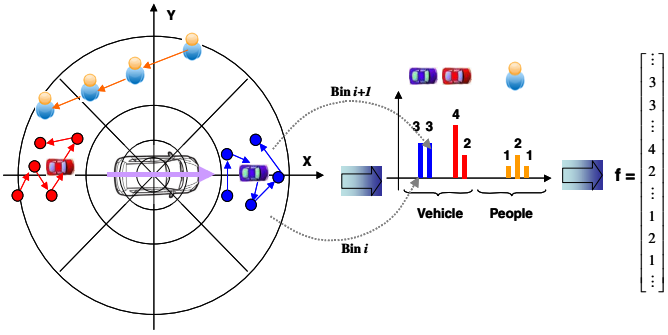


Fig. 4. HO2 computation using log-polar partition function. The reference entity is the middle vehicle in a three-vehicle convoy. The people icons represent a pedestrian crossing the street. The histograms of vehicle and people occurrences are shown in the middle. The resulting space-time context feature vector is shown on the right.

Finally, as already discussed at the third and highest level a finite state machine (FSM) is used to model the training scenario as a set of behavioral states predicated with trigger events (mid-level). The overall structure of the FSM and trigger events is determined by a Training Ontology that is specific to the TTP (techniques tactics and procedures) of the training scenario.

2.3 Trainee Performance Evaluation

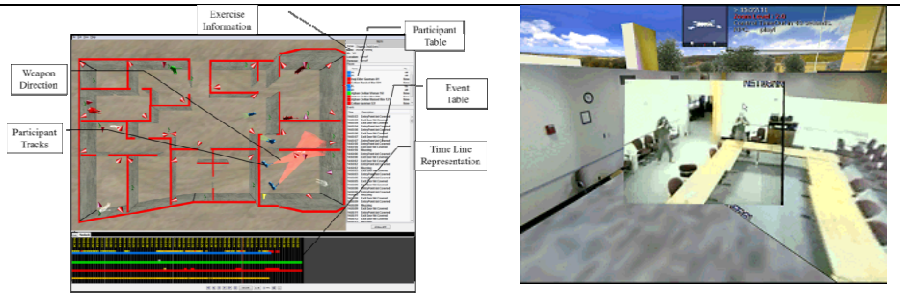
Performance metrics are computed by comparing trainee actions to canonical executions based on the TTP. Our system computes performance metrics associated with each state during a training exercise. Low-level data including location, weapon orientation etc. is used to compute these metrics. For our MOUT application training, the following performance metrics are computed:

- **360 degrees Security:** The percentage of a full 360 degrees that is either covered by a Warfighter's weapon or is blocked by a cover.
- **Blocking:** The fraction of the time that all danger spots were blocked by the warfighters, i.e. at least one warfighter points his weapon at each of the danger spots. The danger spot may be a possible sniper position or an approaching vehicle, etc. We use "Aim Margin" to determine the blocking accuracy which needs to be achieved.

- **Cover:** The fraction of time that all warfighters maintain cover. The source of cover can be natural objects such as trees, ravines, hollows, reverse slopes, etc. or man-made such as vehicles, trenches, and craters." [USMC, 2006]. The Warfighters are maintaining cover if the minimum distance of each warfighter from any of the source of cover against the threat direction is below the 'Cover Margin'. The sources of cover are computed from the 3D-model of the training environment. We use Hausdorff distance as the distance measure.
- **Flagging/Muzzling:** A warfighter points his weapon at a friendly. The Flagging score is the total number of detected flaggings.
- **Dispersion Measure:** The average nearest-neighbor distance (NND) per unit time for the warfighter team. Depending on the context, dispersion measure can be useful for signaling "bunching". For instance, "bunching" may be preferred in an urban context when a unit approaches the corner of a building; it is dangerous in open spaces in a rural context where the entire unit may be exposed.

3 Visualization and AAR

We have developed visualization tools that allow a user to view not only videos captured during an exercise, but also tracks trainees, and events in an interactive and easy-to-use manner. Two displays are provided by our system and they are used simultaneously in a synchronized fashion. They are (1) Symbolic Map Display and (2) Video Flashlight Display.



(a) Symbolic Map Display for AAR (b) Video Flashlight Display for AAR

Fig. 5. Visualization and AAR

3.1 Symbolic Map Display

In Figure 5(a), we show a snapshot of a typical exercise as it is displayed by the Symbolic Map Display. The left side of the Symbolic Map Display shows a 3D-model of the MOUT environment. The Symbolic Map Display allows users to view an exercise at any instant and track movements forward and backward in time. A user can also drag the red time line to any location and play back from there.

3.2 Video Flashlight Display

Our system uses multiple video cameras to cover a MOUT facility and captures the entire exercise. For this purpose we developed the Video Flashlight Display system [Kumar, 2003, Hsu, 2000] that can seamlessly integrate multiple video streams into a unified live display. Each of the videos is projected on the 3D-model of the environment. An example of the Flashlight Video display is show in Figure 5(b), where two video streams are projected onto the 3D-model of the scene.

4 Experiments

Our prototype training system has been tested on mock warfighter training exercises including both indoor room clearing and outdoor patrol scenarios. Two of the outdoor training exercises that we have focused on are “React to IED Threat” and “Patrolling”. Figure 5 shows an illustration of one of the simulated “React to IED Threat” exercises performed by a group of eight trainees. To perform the exercise correctly first the Marine unit needs to move towards the building to obtain cover. Once in cover one team blocks the sniper with their weapons while the other team moves across to engage and neutralize the threat.

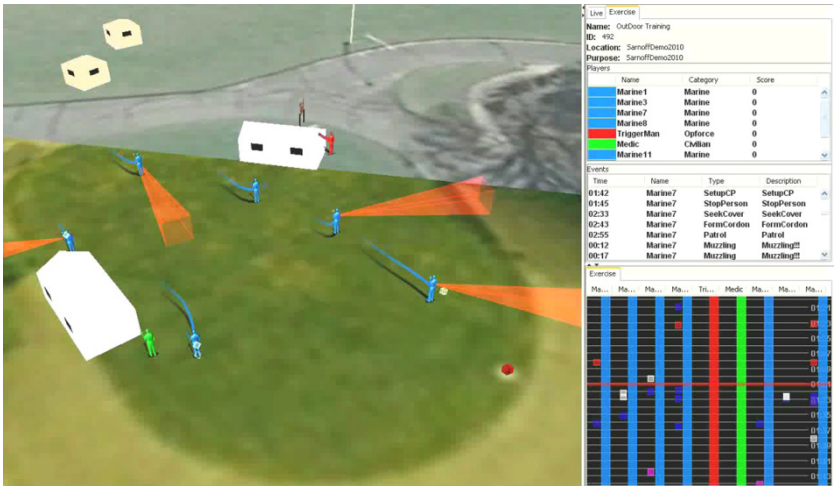


Fig. 6. Bird's eye view illustration of a "React to IED threat" exercise

In Figure 6 we show the metrics computed by our system for the “React to IED” exercise illustrated in Figure 5 (exercise 343). For “Flagging” and “Blocking” we used an “Aim Margin” parameter of 10 degrees. The “Cover Margin” parameter for “Cover Score” was set at 2 meters for this exercise. These values were selected based on empirical testing. A number of observations can be drawn from the plot in Figure 6. Only one “High Dispersion” event was detected indicating that the trainees maintained good nearest neighbor distance. On the other hand there is large number of “Faulty Blocking” and “Faulty Cover” events detected. An interesting comparison is

to view performance of two different teams on the same exercise. In figure 7 we show performance metrics comparing two teams doing the same exercise. Such comparisons are extremely useful in evaluating the impact of training and identify what metrics are more pertinent than others.

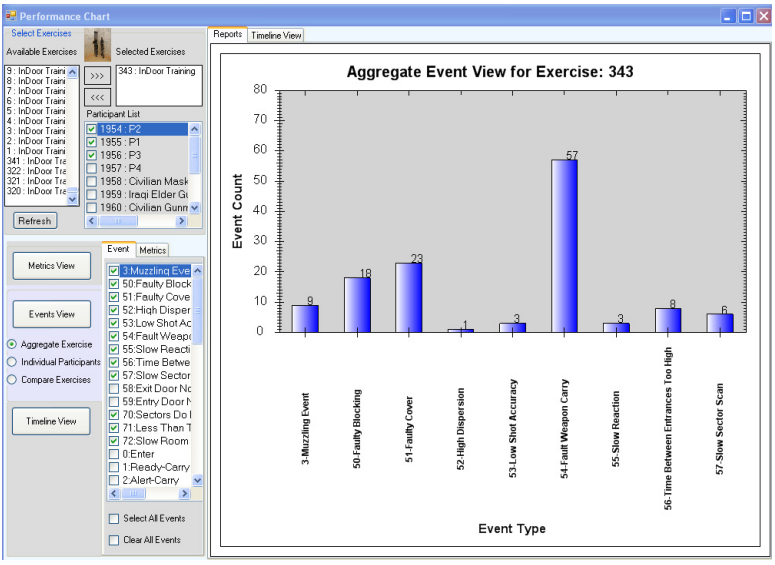


Fig. 7. Performance metrics for an exercise. Events corresponding to metrics like "Muzzling", "Cover" and others are shown.

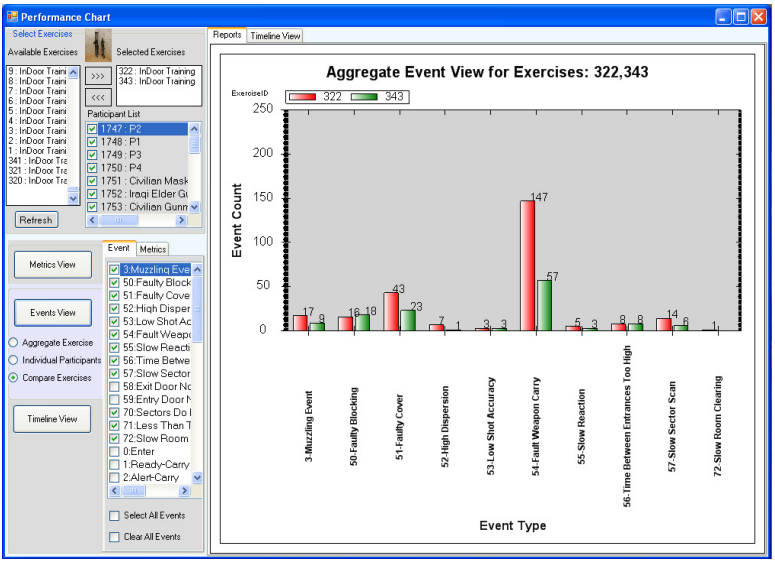


Fig. 8. Comparison between two different teams performing the same exercise

5 Conclusions

We have developed a computational framework for automated behavior analysis and performance evaluation that effectively incorporates TTP and designed training scenarios. Our approach is to use a hierarchical framework that uses a FSM at the top level to capture TTP objectives. Trigger events that transition the state machine from one state of the scenario to another are detected using classifiers on the HO2 feature. To capture trainee behavior, the prototype training system captures and computes tracks, poses and actions of the participants and automatically assesses the performance of warfighters using a training ontology. We have developed a prototype system that has been demonstrated to accurately detect participants' states, mistakes, such as muzzling, automatically. The detected events and computed performance metrics provide power tools for advanced AAR capabilities.

Acknowledgments. This work has been supported by the Office of Naval Research (ONR) program BASE-IT contract N00014-08-C-0127. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of ONR, or the U.S. Government.

References

1. Cheng, H., Yang, C., Han, F., Sawhney, H.: HO2: A new feature for multi-agent event detection and recognition. In: Computer Vision Pattern Recognition Workshop, pp. 1–8 (2008)
2. Hsu, S., Samarasekera, S., Kumar, R., Sawhney, H.S.: Pose Estimation, Model Refinement, and Enhanced Visualization Using Video. In: Proc. IEEE Conf. on Computer Vision and Pattern Recognition, Hilton Head Is., SC, vol. I, pp. 488–495 (2000)
3. Jung, S., Guo, Y., Sawhney, H., Kumar, R.: Action Video Retrieval Based on Atomic Action Vocabulary. In: Proc. ACM Int'l Conf.on Multimedia Information Retrieval, Vancouver, British Columbia (2008)
4. Cheng, H., Kumar, R., Basu, C., Han, F., Khan, S., Sawhney, H., Broaddus, C., Meng, C., Sufi, A., Germano, T., Kolsch, M., Wachs, J.: An Instrumentation and Computational Framework of Automaoted Behavior Analysis and Performance Evaluation for Infantry Training. In: Proceedings of 2009 Interservice/Industry Training, Simulation, and Education Conference (I/ITSEC 2009), Orlando, FL (2009)
5. Cheng, H., Kumar, R., Germano, T., Meng, C.: Automatic Performance Evaluation and Lessons Learned (APELL) for MOUT Training. In: Proceedings of 2006 Interservice/Industry Training, Simulation, and Education Conference (I/ITSEC 2006), Orlando, FL (2006)
6. Kumar, R., Samarasekera, S., Arpa, A., Aggarwal, M., Paragano, V., Hanna, K., Sawhney, H., Sartor, M.: Monitoring Urban Sites using Video Flashlight and Analysis System. In: GOMAC Proceedings, Tampa Florida (2003)

7. Fontana, R.J.: Recent System Applications of Short-Pulse Ultra-Wideband (UWB) Technology. IEEE Transaction on Microwave Theory and Techniques 52(9), 2087–2104 (2004)
8. Noy, N.F., Sintek, M., Decker, S., Crubezy, M., Fergersen, R., Musen, M.A.: Creating Semantic Web Contents with Protégé-2000. IEEE Intelligent Systems 16(2), 60–71 (2001)
9. Melnik, S., Garcia-Molina, H., Papepcke, A.: A Mediation Infrastructure, for Digital Library Services. ACM Digital Libraries, 123–132 (2000)
10. Viola, P., Jones, M.: Robust Real-time Object Detection. In: 2nd Intl Workshop on Statistical and Comp. Theories of Vision, Vancouver (2001)
11. Wachs, J.P., Goshorn, D., Kölsch, M.: Recognizing Human Postures and Poses in Monocular Still Images. In: Intl. Conf. on Image Processing, Computer Vision, and Pattern Recognition (IPCV) (2009)
12. Torralba, S.A., Murphy, K.P., Freeman, W.T.: Sharing visual features for multiclass and multiview object detection. IEEE PAMI 29(5), 854–869 (2007)
13. Camouflage, Cover and Concealment, Lesson Plan. USMC, Weapons and Field Training Battalion (January 26, 2006)
14. Zhao, T., Aggarwal, M., Kumar, R., Sawhney, H.S.: Real-time Wide Area Multi-camera Stereo Tracking. In: Proc. IEEE Conf. on Computer Vision and Pattern Recognition, San Diego, CA (2005)