Monotonicity and Error Type Differentiability in Performance Measures for Target Detection and Tracking in Video

MSR-TR-2012-23

Ido Leichter and Eyal Krupka
Advanced Technology Labs Israel
Microsoft Research

Abstract

There exists an abundance of systems and algorithms for multiple target detection and tracking in video, and many measures for evaluating the quality of their output have been proposed. The contribution of this paper lies in the following: first, it argues that such performance measures should have two fundamental properties – *monotonicity* and *error type differentiability*; second, it shows that the recently proposed measures do not have either of these properties and are thus less usable; third, it composes a set of simple measures, partly built on common practice, that does have these properties. The informativeness of the proposed set of performance measures is demonstrated through their application on face detection and tracking results.

1 Introduction

Multiple target detection and tracking in video is an important research subject in computer vision, and many systems have been developed for this task. A necessary complement to these systems is a means for quantitatively evaluating their performance. Such a quantitative evaluation is necessary for two reasons: enabling a comparison between these systems in terms of performance, as well as enabling the tuning of their parameters for performance optimization. The quantitative evaluation of the performance of a system is done by applying the system on a specific dataset, and then calculating a set of *performance measures* that quantify the quality of the system's output with respect to the dataset's ground truth (GT). Repeating this for multiple systems or for different parameter settings using a common dataset provides a means for quantitatively comparing the systems' performances or finding the optimal parameter setting. Various datasets annotated with GT have been introduced (e.g., [8, 10]), as well as tools for annotating new datasets and for calculating performance measures (e.g., [6, 2, 13]).

The problem of defining performance measures in the context of multiple target detection and tracking is ill-posed in the sense that there is no single "correct" or "best" set of measures. Therefore, many sets of measures have been proposed, many times as a result of multi-party efforts attempting to reach an agreed set of measures. One example of such efforts is the IEEE International Workshop Series on Performance Evaluation of Tracking and Surveillance (PETS) [3]. Another example is the VACE metrics [8], developed in the course of a series of evaluations conducted in the framework of the US Government's Video Analysis and Content Extraction (VACE) program. The CLEAR MOT metrics [5] were developed as part of the CLEAR consortium [4], coordinated by the US National Institute of Standards and Technology (NIST) and by Karlsruhe Institute of Technology. A close variant of the latter metrics is the CLEAR metrics described in [8]. Another recently proposed set of performance measures consists of the pair of information theoretic measures in [7]. An elaborated review of earlier performance measures and additional evaluation programs is provided in [8].

As mentioned, the problem of defining performance measures in the context of multiple target detection and tracking is ill-posed. Nevertheless, we claim that any set of measures should have the following two fundamental properties. The first, termed here *monotonicity*, is that the elimination of an error or the addition of a success should result in each measure in the set being improved or unchanged (an exact definition is provided in Sec. 5.1). The second property, termed here *error type differentiability*, is that the set of measures should be informative about the system's performance with respect to each of the different basic error types. Otherwise, as is explained in Sec. 5.2, the performance measures may not be able to tell which system or parameter setting is better for the application at hand. In fact, if the performance measures are not error type differentiating, it may not be clear what application they may be used for. This paper shows that the recently proposed performance measures do not have either of these two properties and are thus less usable. A set of measures that does have these properties is proposed as well.

The rest of the paper proceeds as follows: Sec. 2 defines the terminology that is being used, Sec. 3 lists the addressed types of error and success, Sec. 4 briefly discusses recently proposed performance measures, the two performance measures' properties argued as being necessary are discussed in Sec. 5, a set of performance measures that has these properties is proposed in Sec. 6 and experimentally tested in Sec. 7, and a conclusion is provided in Sec. 8.

2 Terminology

Throughout the paper the following terminology is used:

Truth target – An instance of a true target object in a frame.

System target – An instance of an object reported by the system in a frame.

Target's location – The values of variables that define the location of a truth or system target. For example, when the target's location is approximated in the frame by an axes-aligned square, these variables may consist of the square's center and length of side. It is assumed here that the annotated locations of the truth targets and the estimated locations of the system targets are in the same state space. In case the spaces are different, they should be reduced to a common one.

Truth Track – The sequence of all truth targets corresponding to a true target object. Note that the frame indices in the track are not necessarily consecutive, as the object may be temporarily occluded or outside the camera's field of view.

System Track – The sequence of all system targets corresponding to an object reported by the system. Similarly to a truth track, the frame indices in a system track are not necessarily consecutive.

GT annotation – The set consisting of all truth tracks.

System's output – The set consisting of all system tracks.

3 Basic Types of Error and Success

There are various ways to partition the possible detection and tracking errors into a set of basic error types. The most common basic types of error addressed in the literature are

1. *False negative* – No system target is associated with a truth target. The performance is degraded as the number of false negatives increases.

2. *False positive* – A system target is not associated with a truth target. The performance is degraded as the number of false positives increases.

3. *Fragmentation* – A truth track contains targets that are associated with different system tracks. The performance is degraded as the number of fragmented truth tracks increases and as these tracks become more severely fragmented.

4. *Merger* – A pair of truth tracks contain target pairs – one target of each track – that are associated with a common system track. The performance is degraded as the number of merged truth track pairs increases and these track pairs become more severely merged.

5. *Deviation* – The system target's location deviates from its associated truth target's location. The performance is degraded as the deviations become greater.

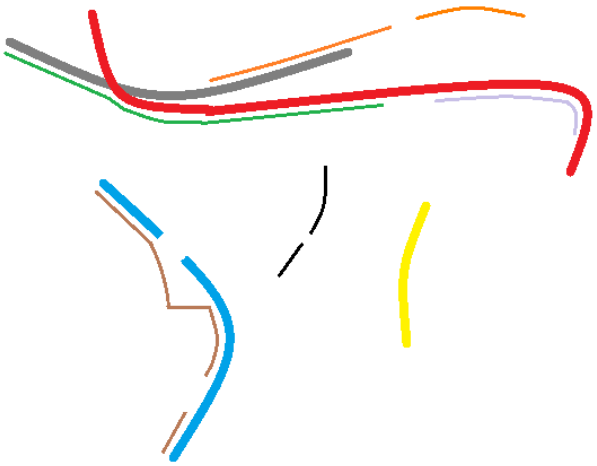

Figure 1: Illustrations of basic error types. Thick (Thin) lines are truth (system) tracks. Lines of the same color are parts of the same track. There are false negatives due to six undetected truth track sections: both ends and middle of the red track, the middle of the gray track, near the lower end of the turquoise track, and the whole yellow track. Four system track sections consist of false positives: the right-hand side section of the orange track, the right-hand side end of the other section of this track, the middle of the brown track, and the whole black track. Truth targets of the gray and the red truth tracks are merged by the green system track. Truth targets of the red truth track are fragmented into the lavender and green system tracks.

The first four basic error types are illustrated in Fig. 1. These error types are basic in the sense that other types of error are combinations of these types. For example, a "gap" [7] is a sequence of consecutive false negatives corresponding to one truth track, an "extra" [7] is a sequence of consecutive false positives in one system track, and an "ID swap" [12] is a combination of fragmentation errors and a merger related to two truth tracks and two system tracks.

Complementary to an error, the lack of a potential error is considered as a *success*. Symmetrically to the detection and tracking error types, the corresponding types of detection and tracking success are:

1. *True positive* – A system target is associated with a truth target. The performance is improved as the number of true positives increases.

2. *True negative* – The lack of a system target that would have been a false positive. The performance is improved as the number of true negatives increases.

3. *Identification* – A truth track contains targets that are associated with the same system track. The performance is improved as the number of identification successes increases and

as the truth tracks corresponding to these successes become less fragmented. Note that, unless the truth track is completely fragmented (i.e., each of the system targets associated with it is of a different system track) or wholly identified as a single identity (i.e., all system targets associated with it are of the same system track), both a fragmentation error and an identification success are associated with this truth track.

4. *Differentiation* – A pair of truth tracks contain target pairs – one target of each track – that are associated with different system tracks. The performance is improved as the number of differentiation successes increases and as the truth track pairs corresponding to these successes become less severely merged. Note that, unless the pair of truth tracks is completely merged (i.e., all the system targets associated with one of the truth tracks are of the same system track) or completely differentiated (i.e., targets of different truth tracks are never associated with a common system track), both a merger error and a differentiation success are associated with this pair of truth tracks.

5. *Proximity* – The system target's location is in proximity to its associated truth target's location. The performance is improved as the proximity between the system targets and their associated truth targets becomes greater. Note that, unless the system target's location matches the location of the associated truth target perfectly, a proximity success constitutes a deviation error as well, and vice versa.

4 Recently Proposed Performance Measures

Several sets of performance measures have been recently proposed: VACE [8], CLEAR [8], CLEAR MOT [5], and the information theoretic measures [7].

The VACE metrics [8] consist of two scores – Sequence Frame Detection Accuracy (SFDA) and Average Tracking Accuracy (ATA). SFDA evaluates the performance with respect to false negatives, false positives and deviation errors; ATA evaluates the performance with respect to all errors.

The CLEAR metrics [8] consist of four scores – Normalized Multiple Object Detection Accuracy (N-MODA), Normalized Multiple Object Detection Precision (N-MODP), Multiple Object Tracking Accuracy (MOTA), and Multiple Object Tracking Precision (MOTP). N-MODP and MOTP evaluate the performance with respect to deviation errors, N-MODA evaluates the performance with respect to false negatives and false positives, and MOTA evaluates the performance with respect to false negatives, false positives, fragmentation, and mergers.

A close variant of the CLEAR metrics is the CLEAR MOT metrics [5]. These metrics consist of the MOTA and MOTP scores. However, in the MOTA score of the CLEAR MOT metrics the penalties for the different errors are fixed, whereas in the CLEAR metrics the penalties are tunable.

In [7], a pair of information theoretic measures was proposed: Truth Information Completeness and False Information Ratio. The former quantifies the percentage of the truth information captured and the latter quantifies the amount of false information generated.

5 Properties that the Set of Measures Should Have

A set of measures that evaluates the quality of the system's output should have the following two properties: *monotonicity* and *error type differentiability*.

5.1 Monotonicity

The elimination of an error or the addition of a success should result in an improvement of the measures, and similarly for the reduction of a deviation or, equivalently, the increase of a proximity. More exactly:

> *If the system's output or the GT annotation is modified such that an error (success) is eliminated (added) or reduced (increased) with no error (success) introduced (eliminated) or worsened (moderated), then each measure should be improved or remain unchanged.*

In particular, each measure should be improved or remain unchanged upon the following:

1. The elimination of false negatives or the addition of true positives.
2. The elimination of false positives or the addition of true negatives.
3. The elimination of fragmentation errors or the addition of identification successes.
4. The elimination of merger errors or the addition of differentiation successes.
5. The elimination or reduction of deviation errors or, equivalently, the increase of the proximities of system targets to their associated truth targets.

Allowing that a measure remains unchanged as well accounts for the case where the type of eliminated (added) error (success) is irrelevant to the measure, or in the boundary case where the error (success) is eliminated (added) when there have not been any successes (errors) of the complementary type beforehand.

5.1.1 Lack of Monotonicity in Previous Measures

The aforementioned performance measures (VACE [8], CLEAR [8], CLEAR MOT [5], and the information theoretic measures [7]) are not monotonic. This is shown in Figs. 2-4, which provide simple, concrete examples that the elimination of errors or the addition of successes may result in the degradation of one or more measures. Even worse, for each set of measures, the error

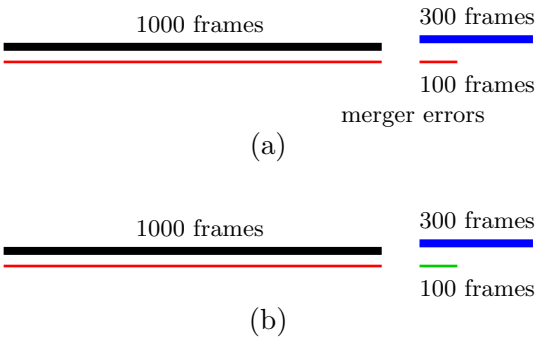

Figure 2: **The VACE metrics [8] are not monotonic.** Thick (Thin) lines are truth (system) tracks. Lines of the same color belong to the same track. Image (a) illustrates a GT annotation consisting of two tracks, as well as one system track. While the locations of the 1000 system targets of the left-hand side section are exact, the other 100 system targets contain deviation errors such that the mean overlap ratio [8] between them and the corresponding section of the blue track is 0.5. Image (b) illustrates the same truth tracks and two system tracks resulting from splitting the former system track into two separate tracks. The splitting of the system track in (a) eliminates the merger errors without introducing any new errors. However, instead of being improved, the scores of the VACE metrics are degraded: the SFDA is not affected (0.808) and the ATA is reduced from 0.606 to 0.583.

eliminations or success additions cause the degradation of one or more of the measures and the improvement of none.

5.2 Error Type Differentiability

The set of measures should provide a performance evaluation with respect to each of the basic error types *alone*. The reason for this requirement is that the severity of the different error types is application dependent. One detection and tracking system may be better than another in one context and worse in another context. For example, in applications that require high recall, such as surveillance, a false negative is typically more severe than a false positive or a deviation; in applications that require high precision such as video summarization for entertainment purposes it is the other way around. As another example, consider face detection and tracking as a preceding process for recognizing the person in each track vs a preceding process for video summarization. In the former case, mergers are more severe than fragmentation as a system track that contains faces of multiple people has no single identity associated with it. In the latter case, the relation between the severities of these two error types is not necessarily the same.

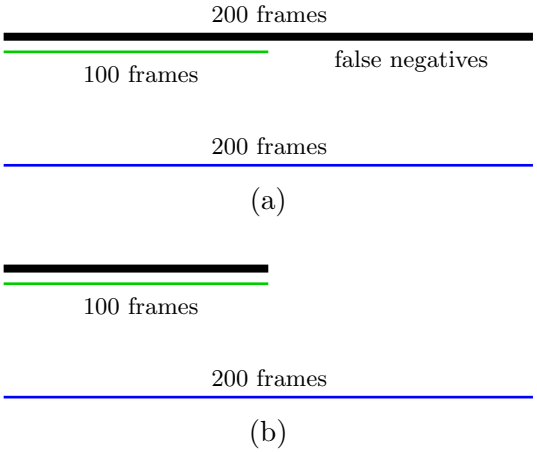

Figure 3: **The CLEAR metrics [8] and the CLEAR MOT metrics [5] are not monotonic.** Thick (Thin) lines are truth (system) tracks. Image (a) illustrates a GT annotation consisting of one track, as well as two system tracks, the green of which corresponds to the truth track. Image (b) illustrates the same three tracks, with the truth track shortened. The shortening of the truth track eliminates 100 false negatives without introducing any new errors. However, instead of being improved, the scores of the CLEAR metrics are degraded: the MOTP and the N-MODP are not affected (let the image sequences in (a) and (b) be of the same length), and the MOTA and the N-MODA are reduced from -0.5 to -1 based on the same error type weighting used in [8]. The same happens under the CLEAR MOT metrics, which consist of the MOTP and MOTA measures.

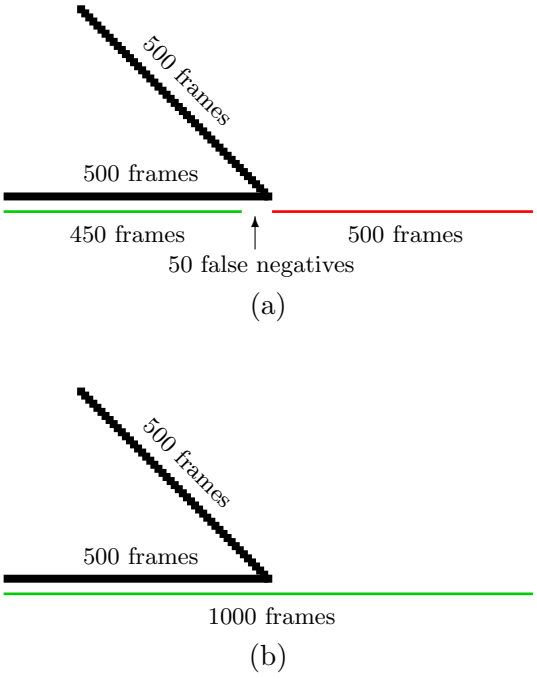

Figure 4: **The information theoretic measures in [7] are not monotonic.** Thick (Thin) lines are truth (system) tracks. Image (a) illustrates a GT annotation consisting of one track, as well as two system tracks, the green of which corresponds to the truth track. Image (b) illustrates the same GT annotation with the green system track lengthened and unified with the other system track. The modifications in (b) to the system's output eliminate 50 false negatives without introducing any new errors. However, instead of being improved, both information theoretic measures are degraded: the truth information completeness and the false information ratio change from 0.4178 and 0.6013 to 0.3982 and 0.6018, respectively. These figures were obtained based on a sequence length of 1000 frames and 10000 *states* [7] per frame.

5.2.1 Lack of Error Type Differentiability in Previous Measures

The VACE metrics were recently proposed in [8]. As mentioned, these metrics consist of only two measures – SFDA and ATA. SFDA is a single score that evaluates the quality of the system's output with respect to false negatives, false positives and deviation errors; ATA is a single score that evaluates the quality with respect to all errors. Since these metrics lack error type differentiability, they usually cannot provide an indication about how good the system is for the application at hand, or which of several systems is better for it. In fact, in the case where the VACE metrics score one system higher than another, it is not very clear for what application the former system is better than the latter. Therefore, while the VACE metrics serve as a compact representation of the system's performance, it is not exactly clear what purpose they may be used for. Furthermore, the collective quantification of the errors of different basic types may render the relation between the measure and the quality of the system's output inappropriate in many cases. Two examples of such a case are shown in Figs. 5 and 6. In [8] it is mentioned that these metrics are "less usable because the measures did not...identify failure components for debugging purposes." We stress that their lack of error type differentiability makes them less usable also because they do not really tell how good the system is for the application.

Other recently proposed performance measures are the CLEAR metrics [8]. As mentioned, these metrics consist of four measures – N-MODA, N-MODP, MOTA, and MOTP. While N-MODP and MOTP indeed evaluate the performance with respect to deviation errors alone, N-MODA is a single score that evaluates the performance with respect to false negatives and false positives, and MOTA is a single score that evaluates the performance with respect to false negatives, false positives, fragmentation, and mergers. Therefore, the CLEAR metrics do not differentiate between most of the error types as well. As mentioned, a close variant of these metrics that has this problem as well is the CLEAR MOT metrics [5]. There, the MOTA measure assigns exactly the same penalty for a false negative in a single frame and for the merger errors resulting from a mismatch error [5] (a *mismatch error* in [5] is a match between a GT ID and a system ID that contradicts the ID matching in previous frames) although the latter error is much more severe than the former in most contexts (see illustration in Fig. 7). In [8], the different errors are weighted by their type. This still does not solve the problem: First, it is not clear how the weighting should be done. In the experiments in that work, false negatives and false positives are assigned the weight of 1, and the number of ID switches is replaced with $\log_{10}(\#\text{ID-switches}+1)$ (an *ID switch* in [8] is the same as a mismatch error in [5] – a match between a GT ID and a system ID that contradicts the ID matching in previous frames). It is not clear how this weighting was decided. In particular, it is not clear why it was decided that the more ID switches the smaller their weight-per-switch (which is a consequence of the log function's decreasing slope), as well as how the specific base of the logarithm was chosen. In addition, this weighting assigns a perfect score to a system's output that consists of mergers or fragmentation errors resulting from one ID switch ($\log_{10}(0+1)=0$). Second, the

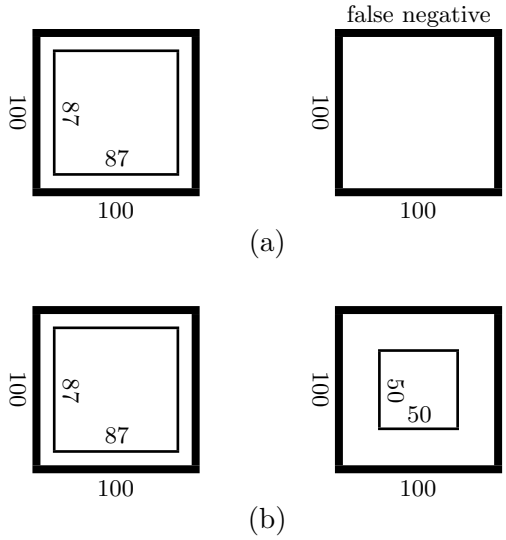

Figure 5: **The frame detection accuracy as defined by the VACE metrics is identical in both cases.** Thick (Thin) boxes are truth (system) targets. Image (a) illustrates a frame consisting of two truth targets, each is a 100x100 square. The system detected the left-hand side target as an 87x87 square and did not detect the right-hand side target. Image (b) illustrates the same frame with both targets detected, the left-hand side target is detected exactly as before and the right-hand side target is detected as a 50x50 square. In (a) there is a false negative whereas in (b) the deviations are greater in average. These two opposite factors cannot be reflected by the VACE metrics, which consist of only one frame-based measure (SFDA). In fact, the frame detection accuracy as defined by the VACE metrics is identical in both cases (0.50), although the quality of the system's output in (b) is considered higher than that in (a) for most applications – in (b) the right-hand side target is inaccurately localized, whereas in (a) it is not detected at all. Any reduction of one the system targets' sizes in (b) will even make this system's output less favorable than that in (a).

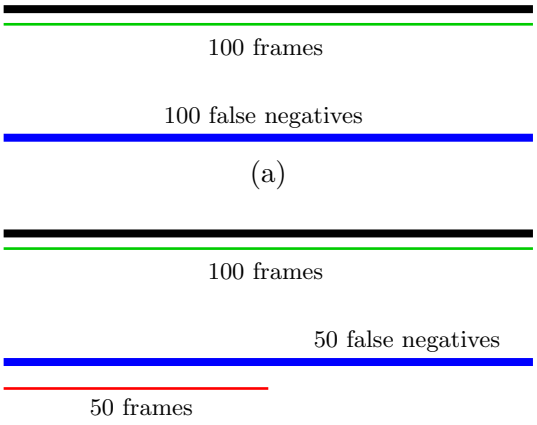

Figure 6: **The VACE metrics favor case (a) over case (b).** Thick (Thin) lines are truth (system) tracks. Image (a) illustrates a GT annotation consisting of two tracks, as well as one system track that matches perfectly with the black truth track (i.e., overlap ratio [8] equals 1). Image (b) illustrates the same three tracks plus the red system track that corresponds to the blue truth track. The red track is inaccurate, which causes its mean overlap ratio with the corresponding half of the blue track to be 0.25. In (b) there are less false negatives but the deviations are greater in average. This is not reflected in the VACE metrics, *both* favoring case (a) over case (b) ((SFDA,ATA)=(0.667,0.667) vs (SFDA,ATA)=(0.646,0.563), respectively). In fact, the preference of case (a) by the VACE metrics contradicts the typical tendency to favor case (b) over case (a) – in (b) the blue target is poorly tracked whereas in (a) it is not tracked at all.

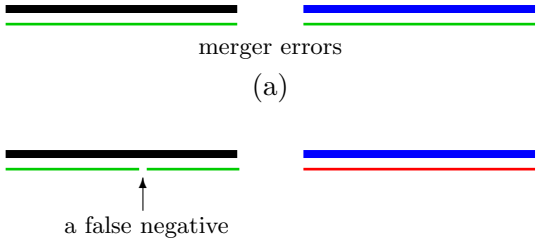

Figure 7: The CLEAR MOT metrics [5] assign the merger errors resulting from a mismatch error (case (a)) and a false negative in a single frame (case (b)) exactly the same penalty, although typically the former mistake is much more severe than the latter. The weighting used in the CLEAR metrics [8] even assigns a mismatch error a smaller penalty than a single false negative. Thick (Thin) lines are truth (system) tracks. Lines of the same color belong to the same track.

fact that the different errors are weighted by their type still does not solve the problem that one cannot know the source errors of the final, weighted score and their distribution.

As mentioned, a pair of information theoretic measures was recently proposed in [7], one quantifies the percentage of the truth information captured and the other quantifies the amount of false information generated. The first measure is affected by false negatives, false positives, and mergers; the second measure is affected by these basic error types as well as by fragmentation. Therefore, although concise, this pair of measures implicitly assign weights to the different types of error and lack error type differentiability as well. A clear illustration of this is provided in Fig. 7 in [7], where it is shown that the pair of measures may be very similar for different basic error types.

6 Monotonic and Error Type Differentiating Performance Measures

Following are provided five error measures, each one measures the quality of the system's output with respect to one of the basic error types (and its complementary success type) listed in Sec. 3. Each measure is nonnegative, zero if and only if there are no errors of its respective type, as well as monotonic. As shown, four measures out of the five may be interpreted as prior probabilities that a specific error of the corresponding type occurs, and the other measure is simply the mean deviation error. A method for obtaining a smaller set of monotonic and error type differentiating measures is discussed in Sec. 8. Note that the set of measures is partly built on common practice. Specifically, the first, second, and fifth measures have been widely used

(e.g., [3]).

Inevitably, the performance measures depend on the specific matching between the truth and system targets. In the following formulation of the measures it is assumed that such a matching has already been done in each frame, and that each matching is one-to-one. Different possible matching schemes, including the specific one used here, are discussed in Sec. 6.8.

A Matlab function that calculates the performance measures of a system's output with respect to a GT annotation is provided.

6.1 Notation

- T – the number of frames in the video. The frame indices are $1, 2, \ldots, T$.
- #truth tracks – the number of truth tracks in the GT annotation. Each truth track is associated with a GT ID$\in \{1, \ldots, \#\text{truth tracks}\}$.
- #system tracks – the number of system tracks in the system's output. Each system track is associated with a system ID$\in \{1, \ldots, \#\text{system tracks}\}$.
- (i, t) – target of (truth or system) track i in frame t.
- $m^{GT}(i,t) : \{1, \ldots, \#\text{truth tracks}\} \times \{1, \ldots, T\} \to \{0, 1, \ldots, \#\text{system tracks}, \emptyset\}$ – truth target presence and matching function:

$$m^{GT}(i,t) = \begin{cases} j, & \text{truth target } (i,t) \text{ is matched to system target } (j,t); \\ 0, & \text{truth target } (i,t) \text{ is not matched to any system target} \\ & \text{although it appears in frame } t; \\ \emptyset, & \text{no target of truth track } i \text{ appears in frame } t. \end{cases} \tag{1}$$

- $m^{S}(j,t) : \{1, \ldots, \#\text{system tracks}\} \times \{1, \ldots, T\} \to \{0, 1, \ldots, \#\text{truth tracks}, \emptyset\}$ – system target presence and matching function:

$$m^{S}(j,t) = \begin{cases} i, & \text{system target } (j,t) \text{ is matched to truth target } (i,t); \\ 0, & \text{system target } (j,t) \text{ is not matched to any truth target} \\ & \text{although it is contained in the system's output;} \\ \emptyset, & \text{the system's output does not contain the system target } (j,t). \end{cases} \tag{2}$$

- $\mathbf{x}^{GT}(i,t)$ – the location of truth target (i,t).
- $\mathbf{x}^{S}(j,t)$ – the location of system target (j,t).
- $d(\mathbf{x}^S, \mathbf{x}^{GT})$ – the distance function between a system target's location $\mathbf{x}^S$ and a truth target's location $\mathbf{x}^{GT}$. It is assumed that the function d is nonnegative and zero if and only if $\mathbf{x}^S = \mathbf{x}^{GT}$.

6.2 Measuring false negatives

The measure of false negatives is defined to be the ratio between their number and the total number of truth targets in all frames:

$$\text{False Negative Rate} = \frac{\left|\left\{(i,t) : m^{GT}(i,t) = 0\right\}\right|}{\left|\left\{(i,t) : m^{GT}(i,t) \neq \emptyset\right\}\right|}. \tag{3}$$

The false negative rate (FNR) approximates the prior probability that a given truth target in a frame is unmatched to any system target. It ranges between 0 (i.e., no false negatives) to 1 (i.e., all truth targets are unmatched). In the pathological case where there are no truth targets at all, this measure is undefined. This is in agreement with the fact that a false negative rate can not be calculated when there are zero positives in the dataset.

6.3 Measuring false positives

The measure of false positives is defined to be their number, normalized by the sequence length and image area A:

$$\text{False Positive Rate} = \frac{1}{T \cdot A}\left|\left\{(j,t) : m^{S}(j,t) = 0\right\}\right|. \tag{4}$$

The false positive rate (FPR) estimates the average number of false positives in a frame per unit image area. It is a nonnegative measure that is 0 if and only if there are no false positives. In statistical analysis, the FPR is defined as the ratio between the number of false positives and the total number of false instances. It thus approximates the conditional probability that an instance is erroneously classified as positive given that it is negative. However, in the context of target detection and tracking in video, the total number false instances is not well defined. This is the reason we normalize this measure by the video "volume" $T \cdot A$, which makes the measure proportional to the prior probability that the system's output erroneously contains a system target in a specific frame and location where a truth target is not present. A similar approach was taken in [7].

6.4 Measuring fragmentation

Given a particular truth track, let the measure of its fragmentation error be the ratio between the number of unordered pairs of targets of this track that are matched to different system tracks and the total number of unordered pairs of matched truth targets of this track. Symbolically, denote the set of all matched targets of truth track i by $\mathcal{M}_i^{GT} = \left\{(i,t) : m^{GT}(i,t) \notin \{0, \emptyset\}\right\}$. Then this measure is

$$\begin{aligned}&\text{Fragmentation\_Index}_i \\ &\quad = \left|\left\{\{(i,t_1),(i,t_2)\} : \{(i,t_1),(i,t_2)\} \in \mathcal{P}_i^{\text{same}}, m^{GT}(i,t_1) \neq m^{GT}(i,t_2)\right\}\right| \Big/ \left|\mathcal{P}_i^{\text{same}}\right|,\end{aligned} \tag{5}$$

where $\mathcal{P}_i^{\text{same}} = \left\{\{(i,t_1),(i,t_2)\} : (i,t_1) \in \mathcal{M}_i^{GT}, (i,t_2) \in \mathcal{M}_i^{GT}, t_1 \neq t_2\right\}$ is the set of all unordered pairs of matched truth targets of truth track i.

The fragmentation index of a truth track approximates the prior probability that a specific pair of matched truth targets of this track are erroneously matched to system targets of different tracks. The measure of fragmentation of the system's output is defined to be a weighted average of the fragmentation indices of all truth tracks. The weight of each track is the length of its matched part:

$$\text{Fragmentation Index} = \frac{\sum\limits_{\substack{i \text{ s.t.} \\ \mathcal{P}_i^{\text{same}} \neq \emptyset}} \left|\mathcal{M}_i^{GT}\right| \cdot \text{Fragmentation\_Index}_i}{\sum\limits_{\substack{i \text{ s.t.} \\ \mathcal{P}_i^{\text{same}} \neq \emptyset}} \left|\mathcal{M}_i^{GT}\right|}. \tag{6}$$

In principle, the fragmentation index of the system's output could have been simply defined as the ratio between the number of unordered pairs of targets of the same truth track that are matched to different system tracks and the total number of unordered pairs of matched truth targets of the same track. This would approximate the prior probability that a specific pair of matched truth targets of the same track is erroneously fragmented when the drawing of the pair is performed uniformly over the set of *all* pairs of matched truth target of the same track. However, this would make the weight of each truth track quadratic in the length of its matched part rather than linear as in (6). In the pathological case where $\forall i \ \mathcal{P}_i^{\text{same}} = \emptyset$ (i.e., $\forall i \ \left|\mathcal{M}_i^{GT}\right| \leq 1$), the measure is undefined. This is in agreement with the fact that, in this case, there are no fragmentation errors neither identification successes.

6.5 Measuring mergers

Given a particular pair of truth tracks, let the measure of their merger error be the ratio between the number of unordered pairs of truth targets – one from each track – that are matched to the same system track and the total number of unordered pairs of matched truth targets – one from each track. Symbolically, the measure of merger of the pair of truth tracks i_1 and i_2 is

$$\begin{aligned}&\text{Merger\_Index}_{\{i_1,i_2\}} \\ &= \left|\left\{\{(i_1,t_1),(i_2,t_2)\} : \{(i_1,t_1),(i_2,t_2)\} \in \mathcal{P}_{\{i_1,i_2\}}^{\text{diff}}, m^{GT}(i_1,t_1) = m^{GT}(i_2,t_2)\right\}\right| \Big/ \left|\mathcal{P}_{\{i_1,i_2\}}^{\text{diff}}\right|,\end{aligned} \tag{7}$$

where $\mathcal{P}_{\{i_1,i_2\}}^{\text{diff}} = \left\{\{(i_1,t_1),(i_2,t_2)\} : (i_1,t_1) \in \mathcal{M}_{i_1}^{GT}, \ \ (i_2,t_2) \in \mathcal{M}_{i_2}^{GT}\right\}$, $i_1 \neq i_2$, is the set of all unordered pairs of matched truth targets – one target of truth track i_1 and one of truth track i_2.

Similarly to the fragmentation index of a truth track, the merger index of a pair of truth tracks approximates the prior probability that a specific pair of matched truth targets – one of each of these two *different* tracks – are erroneously matched to system targets of the *same* track. The measure of mergers of the system's output is defined to be a weighted average of the merger indices of all pairs of truth tracks. The weight of each pair of tracks is the total length of the their matched parts:

$$\text{Merger Index} = \frac{\sum\limits_{\substack{\{i_1,i_2\} \text{ s.t.} \\ \mathcal{P}^{\text{diff}}_{\{i_1,i_2\}} \neq \emptyset}} \left(\left|\mathcal{M}^{GT}_{i_1}\right| + \left|\mathcal{M}^{GT}_{i_2}\right|\right) \cdot \text{Merger\_Index}_{\{i_1,i_2\}}}{\sum\limits_{\substack{\{i_1,i_2\} \text{ s.t.} \\ \mathcal{P}^{\text{diff}}_{\{i_1,i_2\}} \neq \emptyset}} \left|\mathcal{M}^{GT}_{i_1}\right| + \left|\mathcal{M}^{GT}_{i_2}\right|}. \tag{8}$$

As before, the merger index of the system's output could have been simply defined as the ratio between the number of unordered pairs of truth targets of different tracks that are matched to the same system track and the total number of unordered pairs of matched truth targets of different tracks. This would approximate the prior probability that a specific pair of matched truth targets of different tracks is erroneously merged where the drawing of the pair is performed uniformly over the set of *all* pairs of matched truth targets of different tracks. However, this would make the weight of each pair of truth tracks quadratic in the total length of their matched parts rather than linear as in (8). In the pathological case where for all truth track pairs $\{i_1, i_2\}$ $\mathcal{P}^{\text{diff}}_{\{i_1,i_2\}} = \emptyset$ (i.e., there are no pairs of matched truth targets of different tracks), the measure is undefined. This is in agreement with the fact that, in this case, there are no merger errors neither differentiation successes.

Note that this measure as well as the former may be efficiently calculated by counting the number of matches per each system-truth ID pair. Therefore, there is no need to explicitly iterate over all target pairs.

6.6 Measuring deviations

The measure of deviations is simply the mean distance between system targets and corresponding truth targets:

$$\text{Mean Deviation} = \frac{\sum\limits_{(j,t)\in\mathcal{M}^S} d\left(\mathbf{x}^S(j,t), \mathbf{x}^{GT}\left(m^S(j,t),t\right)\right)}{|\mathcal{M}^S|}, \tag{9}$$

where $\mathcal{M}^S = \left\{(j,t) : m^S(j,t) \notin \{0,\emptyset\}\right\}$ is the set of all matched system targets in all frames. In the pathological case where $\mathcal{M}^S = \emptyset$ (i.e., there are no matched system targets at all), the measure is undefined. This is in agreement with the fact that no measure of system targets' deviation can be calculated when there are no matched system targets.

6.7 Fulfillment of monotonicity and error type differentiability

Each basic type of error and success influences its corresponding error measure only. Therefore, the proposed set of performance measures differentiates between errors of the different basic types. The examples in Sec. 5.1.1, which show that the previous measures are not monotonic, exemplify that the proposed set of measures are monotonic: in Fig. 2 the Merger Index reduces from 1 in case (a) to 0 in case (b) and all other measures remain unchanged; in Fig. 3 the False Negative Rate reduces from 0.5 in case (a) to 0 in case (b) and all other measures remain unchanged; in Fig. 4 the False Negative Rate reduces from 0.55 in case (a) to 0.5 in case (b) and all other measures remain unchanged. The monotonicity with respect to each of the basic types of error and success is proved in the Appendix.

6.8 Matching between Truth and System Targets

The performance measures depend on the matching between the truth targets and the system targets. This matching depends on the necessary conditions under which a specific truth target may be considered as being detected by a specific system target. Relaxed conditions generally imply lower false negative and false positives rates, but a higher mean deviation. Strict conditions generally imply the opposite. These necessary conditions limit the set of system targets that may be matched with a truth target in a frame, and vice versa. However, there may still be multiple legitimate one-to-one matchings in a frame.

Various schemes may be used to choose among the different legitimate matchings. Each scheme may produce a different matching, which may result in different performance measures. Moreover, some matching schemes are biased with respect to one or more performance measures. For example, in the VACE metrics [8], two matchings are produced: a frame-level matching that maximizes the SFDA measure, and a track-level matching that maximizes the ATA measure. In the CLEAR MOT metrics [5], the matching in each frame is generated via a special procedure that favors ID matching consistency with previous frames, few false negatives and false positives, and a small sum of deviation errors. In [7], the locations of the truth and system targets were represented as Gaussian probability density functions, the association costs between truth and system targets were based on normalized Mahalanobis distances, and the matching, which was based on Bayesian probability theory, was generated through a linear assignment algorithm.

Here, the matching between the set of truth targets and the set of system targets in each frame is generated as follows:

1. All system-truth target pairs that satisfy the necessary conditions for being matched are identified.

2. The distance d between the truth and system targets in each identified pair is calculated.

3. The maximum bipartite matching[1] between the truth and system targets that has the smallest sum of distances is calculated.

The maximum matching in Step 3 can be calculated efficiently as follows: 1. Augment the smaller of the two sets (truth targets and system targets) with imaginary targets so that the two sets will be of equal size. 2. Assign a very large distance (larger than the sum of all other distances) between each pair of truth and system targets that cannot be matched or that contain an imaginary target. 3. Solve the corresponding linear assignment problem (e.g., by the Hungarian algorithm).

The above matching procedure resembles that in [8], although it is not equivalent. In [8], the returned matching is that of maximum total spatial overlaps, which is not necessarily a maximum matching.

7 Experiments

7.1 Tested System and Dataset

To test the proposed set of measures, they were employed to measure and compare the quality of the results produced by various operational modes of an in-house developed offline face detection and tracking system. Given a video, this system detects faces in it and tracks them through the video. In its default operational mode ("Mode 0"), the system sequentially executes the following main steps:

1. Shot boundary detection.

2. For each detected shot:

   (a) Face detection in all frames. The face detector is an implementation of the Viola-Jones face detector [11] that was trained for detecting faces in poses (yaw rotation) that range between frontal and profile. In order to detect roll rotated faces, the face detector in Mode 0 is applied on images rotated by ± 30 degrees as well.

   (b) Agglomerative clustering of the detected faces into system tracks. The clustering is based on spatiotemporal proximity and face size similarity.

   (c) Tracking the location of the last (first) face of each system track in the forward (backward) temporal direction. When a tracked face's location overlaps the location of the first (last) face of another system track, this tracking is terminated, and the extended former track and the latter track are merged as they are likely of the same identity. If the tracking of a specific face terminates before such a merging occurs

[1] A *maximum bipartite matching* in a bipartite graph is a matching that consists of the maximum possible number of matches.

(due to low confidence or due to reaching a shot boundary), the corresponding system track's extension is discarded as it may be false. The tracking is accomplished by a color-based, general object tracker [9].

The dataset used for testing was Episode 1 of the first season of the TV series "Coupling," whose annotation is available at [1].

7.2 Technical Details

The annotation provided in [1] consists of the shot boundaries, as well as the center coordinates of the six main actors' faces in each frame, along with the face states: not present, frontal, right profile, left profile, occluded (by another person or an object), or self-occluded (the head is visible but the person turns away from the camera). The tested system is expected and attempts to maintain the same track ID after the face was temporary occluded or outside the camera's field of view (but in the same video shot). Therefore, one truth track was generated per annotated actor and shot where the actor appears. The truth track consisted of all the targets (i.e., the actor's face appearances) in the shot that were in the frontal, profile, and self-occluded states. Note that the faces in the self-occluded state were included in the truth track because the tested system is expected and attempts to track the head during these video sections. The tested system approximates the location of a target (i.e., a face) by a square and thus provides its center coordinates and size. As mentioned, in the annotation of the dataset only the center coordinates of the targets were provided. Therefore, the distance function between a system target's location and a truth target's location was set to be the Euclidean distance between their centers, measured as the fraction of the corresponding square's side length. A matching between a truth target and a system target was allowed under the necessary condition that the above distance was not greater than 0.5.

Since only the annotations of the six main actors were provided, the final image sequence used for the testing consisted of the concatenation of the video shots where only these actors appear. This summed up to a total length of about 18 minutes, with 604 truth tracks consisting of 45,502 targets. In principle, the tested system objective might be defined such that cases where the target is occluded or outside the camera's field of view should be handled up to a maximal duration, or even not handled at all. For such systems, the generated GT annotation would have consisted a larger number of truth tracks of a shorter mean length.

7.3 Operational Modes and Results

The performance measures obtained for the aforementioned dataset under various operational modes (Modes 1–5) were calculated and compared to those obtained under Mode 0 (described above in Sec. 7.1). In the calculation of FPR (4), the image area in this dataset is taken as the unit area size (i.e., $A = 1$). Thus, for this dataset FPR equals the mean number of false positives per frame. The results are summarized in Table 1 and analyzed in what follows.

In Mode 1, Step 2(c), which contains the tracking by the general object tracker, is skipped over. On one hand, this results in fewer detected truth targets, and the FNR in Mode 1 is indeed degraded as compared to that in Mode 0. On the other hand, this results in fewer false system targets as well, and the FPR in Mode 1 is indeed improved. Skipping all the system track mergers in Step 2(c) results in much higher fragmentation, and the fragmentation index in Mode 1 is indeed significantly degraded as compared to that in Mode 0. As some of the track mergers in Step 2(c) are erroneous, skipping them reduces the number of merger errors. Thus, the merger index in Mode 1 is indeed improved as compared to that in Mode 0. Another consequence of skipping Step 2(c) is that the mean deviation is improved. This results from the lack of the system targets generated by the color-based general tracker, whose target localization is less accurate than that of the face detector.

In Mode 2, the shot boundary detection (Step 1) is skipped over and the entire video is treated as a single shot. This results in tracks erroneously spanning multiple video shots, and the merger index is indeed significantly degraded as compared to that in Mode 0. As mentioned, if during the tracking of a specific face in Step 2(c) a shot boundary is reached, the corresponding system track extension is discarded as it may be false. In Mode 2, there are no detected shot boundaries that may cause the discard of these false system targets. Thus, the FPR is indeed significantly degraded as compared to that in Mode 0. On the other hand, a fraction of these non-discarded system targets are in fact true, and the FNR in Mode 2 is indeed marginally better than that in Mode 0. All these non-discarded, true system targets were obtained by the general object tracker, which provides less accurate localization than that provided by the face detector. Thus, The mean deviation in Mode 2 is indeed marginally degraded as compared to that in Mode 0.

In Mode 3, in order to reduce the runtime, the face detector in Step 2(a) is fully applied on every tenth frame only. In other frames, it is only applied in locations and scales close to faces detected in nearby frames. Thus, the set of system targets in this mode lacks part of those in Mode 0. Some of the lacking system targets are false and the others are true. This is indeed reflected by the FNR and FPR differences between the two operational modes. We see that a positive consequence of the elimination of part of the detected faces in Mode 3 is a reduction in fragmentation errors in this dataset.

In Mode 4, the face detector is applied on unrotated images only. As in Mode 3, the set of system targets in this mode lacks part of the those in Mode 0, which results in the same consequences as before. Moreover, although in this operational mode the face detector is applied on unrotated images only, part of the rotated faces are still detected. However, the localization of these detected rotated faces is less accurate than that in Mode 0, where the "full" detector is applied. This results in a moderate degradation of the mean deviation.

In Mode 5, the face detector's thresholds were lowered to minimum. As can be seen, this resulted in a marginal reduction in the number of false negatives, and in increases in the number of false positives and fragmentation errors. The additional true positives are low-confidence face detections returned by the face detector. Such low-confidence detections tend to be less

	FNR	FPR	Frag.	Merg.	Dev.
Mode 0	0.322	0.041	0.009	3.92E-4	0.126
Mode 1	0.374	0.028	0.093	2.97E-4	0.120
Mode 2	0.292	0.303	0.010	4.93E-3	0.132
Mode 3	0.344	0.029	0.006	4.26E-4	0.126
Mode 4	0.348	0.028	0.007	4.31E-4	0.129
Mode 5	0.314	0.100	0.017	3.83E-4	0.128

Table 1: The performance measures obtained for different operational modes of the tested system. Frag. and Merg. stand for Fragmentation Index and Merger Index, respectively, and Dev. stands for Mean Deviation. See text for details.

accurate in location, which is reflected in the marginal degradation of the mean deviation in this operational mode as compared to that in Mode 0.

8 Conclusion

Two important properties that any set of performance measure should have are monotonicity and error type differentiability. It was shown that the recently proposed measures do not have either of these properties. In addition, a set of five intuitive measures that does have these properties and that is partly built on common practice was proposed. Four measures may be interpreted as prior probabilities that a specific error of the corresponding type occurs, and the other measure is simply the mean deviation error. Of course, other sets of performance measures that have these properties could be defined.

The proposed set of measures was composed for the preceding specific types of error and success. For other sets of error and success types, different measures may be defined. In particular, it might be desired to have a *smaller* set of measures. This may be accomplished by: 1. unifying subsets of the basic error and success types between which the differentiation is less important for the application at hand; 2. measuring the performance with respect to a unified error type by the (possibly weighted) sum of the corresponding subset of measures. The obtained smaller set of measures will remain monotonic and error type differentiating with respect to the new set of more general error types.

Finally, it is important to clarify that the two properties – monotonicity and error type differentiability – cannot serve as sufficient conditions for appropriateness of a set of performance measures. These two properties, however, are necessary conditions for appropriateness that are applicable in other contexts as well.

Appendix

Following we prove the monotonicity with respect to each of the basic types of error and success as defined in Sec. 5.1.

False negatives and true positives affect the False Negative Rate measure (3) only. This measure is monotonically increasing in the number of false negatives and monotonically decreasing in the number of true positives. Therefore, the set of performance measures is monotonic with respect to false negatives and true positives.

Deviations and, equivalently, proximities affect the Mean Deviation measure (9) only. Decreasing a deviation or, equivalently, increasing a proximity decreases this measure. Adding a proximity success without introducing a new error and, in particular, without introducing a new deviation (i.e., $d(\mathbf{x}^S, \mathbf{x}^{GT}) = 0$) may only decrease the measure. Eliminating a deviation by removing a deviated system target without removing an existing success and, in particular, without removing a true positive is not possible. These imply that the set of measures is monotonic with respect to deviations and proximities.

False positives and true negatives affect the False Positive Rate measure (4) only. This measure is monotonically increasing in the number of false positives, and therefore the set of performance measures is monotonic with respect to this kind of error. Adding true negatives may only be accomplished by 1. augmenting the video with additional frames or image area, or 2. removing unassociated truth targets from the GT annotation. The first option decreases the False Positive Rate measure, and the second option does not affect it. Therefore, this measure is monotonic with respect to true negatives as well.

Fragmentation errors and identification successes affect the Fragmentation Index (6) only. Define the severity of the fragmentation error associated with a truth track to be the track's fragmentation index (5) and the degree of associated identification success as 1 minus this fragmentation index. Note that the weight of each fragmentation error and identification success in the overall fragmentation index (6) is the number of matched targets in the corresponding truth track. Reducing a fragmentation error or its weight without increasing other fragmentation errors or their weights, without introducing new errors, without reducing identification successes or their weights, and without removing any success, may only be done by the following ways: 1. changing IDs of system targets in a manner that a truth track's fragmentation index is reduced, or 2. changing IDs of truth targets in a manner that totally fragmented truth tracks are shortened or eliminated and non-fragmented truth tracks are extended or formed. Such changes may only result in the reduction of the overall fragmentation index. Therefore, the set of performance measures is monotonic with respect to fragmentation errors. Adding an identification success or increasing the degree or the weight of one without reducing other successes or their weights, without increasing any fragmentation error or its weight, and without introducing new errors, may only be done by the previous ways or by the following: 1. extending the matched part of a non-fragmented truth track in a manner that it remains non-fragmented, or 2. adding a new non-fragmented truth track. As before, such changes may only result in the reduction

of the overall fragmentation index, and therefore the set of performance measures is monotonic with respect to identification successes as well.

Merger errors and differentiation successes affect the Merger Index (8) only. Define the severity of the merger error associated with a pair of truth tracks to be their merger index (7) and the degree of associated differentiation success as 1 minus this merger index. Note that the weight of each merger error and differentiation success in the overall merger index (8) is the number of matched targets in the corresponding pair of truth tracks. Reducing a merger error or its weight without increasing other merger errors or their weights, without introducing new errors, without reducing differentiation successes or their weights, and without removing any success, may only be done by the following ways: 1. changing IDs of system or truth targets in a manner that a merger index of a truth track pair is reduced without changing the total length of the matched parts of each truth track pair, or 2. changing IDs of truth targets in a manner that totally merged pairs of truth tracks are shortened or eliminated and perfectly differentiated pairs of truth tracks are extended or formed. Such changes may only result in the reduction of the overall merger index. Therefore, the set of performance measures is monotonic with respect to merger errors. Adding a differentiation success or increasing the degree or the weight of one without reducing other successes or their weights, without increasing any merger error or its weight, and without introducing new errors, may only be done by the previous ways or by the following: 1. extending the matched part of a perfectly differentiated pair of truth tracks in a manner that this pair remains perfectly differentiated, or 2. adding new truth and system tracks that yield new perfectly differentiated pairs of truth tracks. As before, such changes may only result in the reduction of the overall merger index, and therefore the set of performance measures is monotonic with respect to differentiation successes as well. ■

References

[1] M. Fischer's website at Computer Vision for Human-Computer Interaction Lab, Karlsruhe Institute of Technology. `http://cvhci.anthropomatik.kit.edu/~mfischer/research/person-reidentification/`.

[2] ViPER: The Video Performance Evaluation Resource. `http://viper-toolkit.sourceforge.net/`.

[3] *First–Thirteenth IEEE International Workshop on Performance Evaluation of Tracking and Surveillance (PETS)*, 2000–2010.

[4] *Classification of Events, Activities and Relationships (CLEAR) Evaluation and Workshop*, 2006–2007. `http://www.clear-evaluation.org/`.

[5] K. Bernardin and R. Stiefelhagen. Evaluating multiple object tracking performance: the CLEAR MOT metrics. *EURASIP Journal on Image and Video Processing*, 2008. Article ID 246309.

[6] R. Collins, X. Zhou, and S.K. Teh. An open source tracking testbed and evaluation web site. *PETS*, 2005.

[7] E.K. Kao, M.P. Daggett, and M.B. Hurley. An information theoretic approach for tracker performance evaluation. *ICCV*, pages 1523–1529, 2009.

[8] R. Kasturi, D. Goldgof, P. Soundararajan, V. Manohar, J. Garofolo, R. Bowers, M. Boonstra, V. Korzhova, and J. Zhang. Framework for performance evaluation of face, text, and vehicle detection and tracking in video: data, metrics, and protocol. *PAMI*, 31(2):319–336, 2009.

[9] I. Leichter, M. Lindenbaum, and E. Rivlin. Tracking by affine kernel transformations using color and boundary cues. *PAMI*, 31(1):164–171, 2009.

[10] S. Oh, A. Hoogs, A. Perera, N. Cuntoor, C. Chen, J.T. Lee, S. Mukherjee, J.K. Aggarwal, H. Lee, L. Davis, E. Swears, X. Wang, Q. Ji, K. Reddy, M. Shah, C. Vondrick, H. Pirsiavash, D. Ramanan, J. Yuen, A. Torralba, B. Song, A. Fong, A. Roy-Chowdhury, and M. Desai. A large-scale benchmark dataset for event recognition in surveillance video. *CVPR*, pages 3153–3160, 2011.

[11] P. Viola and M.J. Jones. Robust real-time face detection. *IJCV*, 57(2):137–154, 2004.

[12] F. Yin, D. Makris, and S.A. Velastin. Performance evaluation of object tracking algorithms. *PETS*, pages 17–24, 2007.

[13] J. Yuen, B. Russell, C. Liu, and A. Torralba. LabelMe video: Building a video database with human annotations. *ICCV*, pages 1451–1458, 2009.