This paper was published at the International Conference on Computer Vision, Santiago, Chile, 13–16 November 2015.

Exploring Causal Relationships in Visual Object Tracking

Karel Lebeda Simon Hadfield Richard Bowden
University of Surrey, Guildford, GU2 7XH, United Kingdom
{K.Lebeda,S.Hadfield,R.Bowden}@Surrey.ac.uk

Abstract

Causal relationships can often be found in visual object tracking between the motions of the camera and that of the tracked object. This object motion may be an effect of the camera motion, e.g. *an unsteady handheld camera. But it may also be the cause,* e.g. *the cameraman framing the object. In this paper we explore these relationships, and provide statistical tools to detect and quantify them; these are based on transfer entropy and stem from information theory. The relationships are then exploited to make predictions about the object location. The approach is shown to be an excellent measure for describing such relationships. On the VOT2013 dataset the prediction accuracy is increased by 62 % over the best non-causal predictor. We show that the location predictions are robust to camera shake and sudden motion, which is invaluable for any tracking algorithm and demonstrate this by applying causal prediction to two state-of-the-art trackers. Both of them benefit, Struck gaining a 7 % accuracy and 22 % robustness increase on the VTB1.1 benchmark, becoming the new state-of-the-art.*

1. Introduction

Causality is a relation between two events, a *cause* (source) and an *effect* (consequence). In general terms, we say that an event causes another event (its effect), if it precedes the effect in time and it increases the probability of the effect happening. Although causality has been studied by philosophers for millennia, it received little attention from the scientific community before the twentieth century. Recently, theoretical advances have brought practical progress in the analysis of time series in many scientific areas.

An example of a causal relationship, which can be observed (and exploited) in the area of computer vision, is the relationship between the motions of the camera and an object in Visual Object Tracking (VOT). There are different possible causal relationships. For instance, the motion of the camera instantly causes motion of the object in the image frame. An abrupt movement of the camera (*e.g.* a shake) can cause a tracker to fail even in otherwise simple tracking scenarios. A particular example can be seen in the performance of all submitted trackers on the BICYCLE sequence in the ICCV VOT Challenge 2013 [25]. While this sequence is relatively easy to track in general, there are two challenging moments (see Figure 1, showing the numbers of failed trackers in the VOT Challenge). Many tracking failures are present around frame 180, caused by a strong occlusion, and around frame 140, stemming from an abrupt camera shake. If these were detected and accounted for, many of the failures could be prevented, regardless of the tracker.

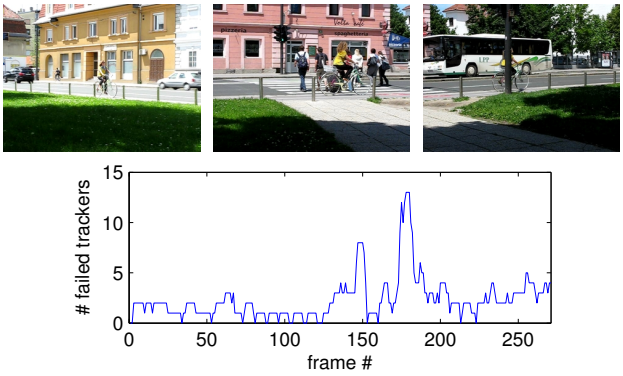


Figure 1. Top: selected frames of the BICYCLE sequence in the VOT Challenge (1, 140&173). Bottom: number of trackers from the challenge, which failed on particular frames. Notice the two challenging moments, a strong occlusion around frame 180 and an abrupt camera shake around frame 140.

Another interesting causal relationship often arises when the motion of the object causes changes of the camera motion. If there is a human in the loop, *e.g.* a cameraman, they are partially tracking an object by definition. A similar conclusion would hold for an automatically-controlled camera, tracking the object. When the object moves towards the edge of the image, the cameraman is likely to move the camera such that the object does not disappear from the scene. An extreme case of this is the satirical Zero-order Tracker [27], shown to successfully track a challenging sequence by simply returning a bounding box on a constant location in the image. As illustrated in Figure 2, here the cameraman kept the diver in the centre of the image frame for almost whole sequence. However, even in less extreme

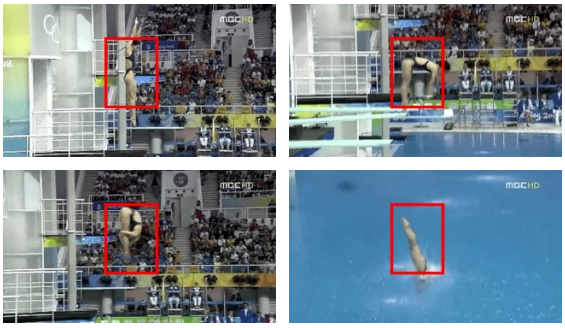


Figure 2. Selected frames from the DIVING sequence, challenging for many trackers, with overlaid "results" of the Zero-order Tracker [27].

cases, the commonly assumed centre bias can be detected, measured and exploited.

It should be emphasised, that our work does not *assume* any kind of high-level oracle (e.g. a human operator) driving the camera motion. In cases where a relationship exists, we can discover and measure its influence. However, if none is present, no causal relationship is found and no further action (such as object motion prediction) is performed. This even extends to changes in behaviour within a sequence. This means the approach can be applied to any existing tracking framework to improve results. To demonstrate this, we look at its effect on two state-of-the-art trackers FoT [28] and Struck [17] on two benchmark datasets. It should be noted that while the prediction helps the tracker, it does not replace it and the result is thus still limited by the abilities of the tracker.

The contributions of this paper are as follows. After describing previous uses of causality and areas of computer vision where the causal relationships are likely to be helpful in Section 2, we show how to measure causality in an information-theoretic manner (which to our knowledge has not been used in any previous work in computer vision) and how to find the properties of the relationships found: if there is a causal relationship, what type (*i.e.* translation, scale,...) and direction (cause vs. effect), and what is the time delay (Section 3). Section 4 shows how these relationships can be used to give prior information to trackers processing the sequence, using two different prediction techniques. This is then experimentally evaluated in Section 5. Finally, Section 6 summarises our contributions and findings.

2. Related work

There have been numerous philosophical publications on causality in both ancient and modern times, originating from Aristotle [1] and significantly influenced by Hume [20]. We take the liberty to omit more recent philosophical publications here, for a recent overview see [30].

One of the early uses of causality for time series analysis was done by Granger [13]. He proposed a statistical causality test, determining the presence of causal relationships between two normally distributed time-series. This approach has become known as *Granger causality* and has been successfully used in economics [18, 35], neurosciences [11, 16], and recently in computer vision [31, 39]. Although it has been revised and improved over the decades (*e.g.* Hacker and Hatemi-J [15] avoided the assumption of a normal distribution), Granger causality is suitable only for linear signals, since it is based on linear regression.

More recently, a novel concept of measuring causality has been proposed: *transfer entropy* (TE), by Schreiber [33]. TE has since found its place in many areas, including again neurosciences [36], chemistry [3] and others [23]. To our knowledge, it has not been previously used in computer vision. As the name suggests, it is based on information theory and is therefore able to detect arbitrary non-linear relationships. In this work, we use TE to measure causation, capturing possibly complex relationships between the motion of the camera and the object.

As previously noted, causal relationships have been examined in the area of computer vision as well. Fan *et al.* [9] used Granger causality to explore actions and temporal dependencies between them in a surveillance scenario. This is then used to cluster and classify video-clips, according to the actions present. In a similar direction is work learning causal relationships between events in video-sequences, which has a potential in action recognition and related tasks. An example is Fire and Zhu [10], who use *Causal And-Or Graphs* and Bayesian grammar models for inference about hidden effects, otherwise undetected, or Sumioka *et al.* [34], using causality to learn joint attention for robots. The work of Brand [4] explores the causal physics of the scene (how mechanics of objects influence other objects).

Prabhakar *et al.* [31] use Granger causality on sequences of keywords directly for the task of human action recognition. Yi and Pavlovic [39] perform the same task, but based on motion-capture data. They use Granger causality to infer the edges in a joint-influence graph of the human body, which improves the performance compared to fixed graphs. Finally, Narayan and Ramakrishnan [29] remove the need for motion-capture systems, using causal relationships between clusters of dense trajectories.

However, to our knowledge there has been no previous work in the field of computer vision exploiting the modern TE approach to causality estimation, and no use of causality for visual object tracking, which is the main application domain for this paper. Learned causal relationships between the motion of the camera and of the tracked object can significantly help a tracker not only to improve accuracy, but to support it during challenging events in the scene. If we were able to estimate a distribution for the object position in the current frame, based on the trajectory of the camera, this

could be supplied to the tracker as prior information. An obvious scenario is *tracking by detection* [2, 12, 17], a popular method for visual object tracking. The tracking is treated as a classification task, where image patches are sampled around the previous location and classified into one of the object/background classes. This can be formulated as a task of maximising the posterior probability of the object pose, where the classification score is a likelihood (given appearance). A prior probability, given by a causality-based prediction, is a very natural complement to this formulation.

On the other side of the spectrum of trackers are *coupled-layer trackers* [5, 26, 38], composed usually from a lower layer of independently tracked features (tracklets) and a higher layer, modelling object shape, motion, etc. The higher layer manages creation of new features such that they are likely to lay on the object, using a soft segmentation mask. This mask can be again enhanced, using prior information about the object motion, provided by a causality-based prediction.

3. Measuring the causal relationships

As mentioned previously, we use transfer entropy as a measure of causality between the camera and the object motion, a formulation employing (differential) entropies (see Sections 3.1 and 3.2). Then we employ a statistical significance analysis (Section 3.3) to discover if the relationship is significant. This is executed in each frame, until such a relationship is found. In the case it is not, we conclude that the motions are unrelated (static camera or independent motion) and we supply no information to the tracker (uniform prior), possibly until the end of the sequence. In the case where a statistically significant causal relationship is found, its parameters are estimated (Section 3.4). We then use this information to predict future object motion (Section 4) and we supply this information to the tracker.

It should be noted that it is impossible to reason about true causality without higher, semantic understanding of the scene. Therefore we work with *predictive causality* instead, which reasons about apparent causal links instead of true causation.

3.1. Differential Entropy

Histogram-based methods are usually employed to estimate the entropy of a random process [6, 7]. However, in our scenario this has two major disadvantages. Firstly, there is an arbitrary choice of bin size for the histograms (for quantisation of continuous signals). Secondly, the number of bins grows exponentially with the number of dimensions. This causes the histograms to be very sparse (and thus not representative of the distribution) and furthermore it requires immense computational cost even for a small number of bins per dimension.

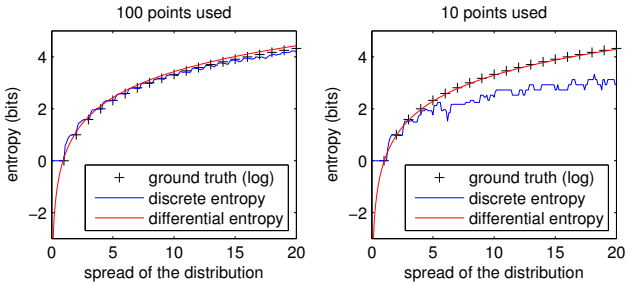


Figure 3. Comparison of discrete and differential entropy. Points were uniformly sampled from an interval $[0;x]$. The histogram bins for discrete entropy computation were fixed at integer positions. Notice how stable differential entropy is, even with sparsely distributed points (no interpolation used).

Instead, we use differential entropy, which operates directly on the continuous variables (see Figure 3 illustrating the advantages of differential entropy). In this work we use the Kernel Density Estimation (KDE, [19]) approach to compute differential entropy, which only requires a choice of kernel (we use a Gaussian kernel with full covariance). The differential entropy of a continuous random process X is

$$H(X) = -\int_X p(x)\log p(x)\,\mathrm{d}x\,, \tag{1}$$

similar to its discrete counterpart. For a finite sample set $\mathcal{S}$ it is approximated using KDE by:

$$\begin{aligned}\hat{H}(X) &= -\frac{1}{|\mathcal{S}|}\sum_{x_i\in\mathcal{S}}\log\hat{p}(x_i)\\ &= -\frac{1}{|\mathcal{S}|}\sum_{x_i\in\mathcal{S}}\log\left(\frac{1}{|\mathcal{S}|-1}\sum_{x_j\in\mathcal{S}\setminus x_i}\kappa_\Sigma(x_i-x_j)\right),\end{aligned} \tag{2}$$

where κ_Σ is a Gaussian kernel with covariance Σ (estimated from the data using expectation maximisation). The probability $p(x_i)$ outside the logarithm is approximated by the distribution of the samples from $\mathcal{S}$ (*i.e.* assuming $\mathcal{S}$ was drawn according to $p(x)$).

3.2. Transfer entropy

Transfer entropy is a measure of directed influence flow between two processes ($X \to Y$, with windows[1] of length n and lag Δt). For continuous signals we define it as:

$$\begin{aligned}T_{X\to Y} = \iiint & p(y_t, \mathbf{y}_t^n, \mathbf{x}_{t-\Delta t}^n)\cdot\\ & \log\frac{p(y_t|\mathbf{y}_t^n, \mathbf{x}_{t-\Delta t}^n)}{p(y_t|\mathbf{y}_t^n)}\,\mathrm{d}y_t\,\mathrm{d}\mathbf{y}_t^n\,\mathrm{d}\mathbf{x}_{t-\Delta t}^n\,,\end{aligned} \tag{3}$$

with time windows defined as $\mathbf{y}_t^n = (y_{t-n}, ..., y_{t-1})$. It can be reformulated (using differential entropies) as the differ-

[1] The window lengths do not necessarily need to be equal for X and Y.

ence of two information gains:

$$T_{X\to Y} = \left(H(Y_t, \mathbf{Y}_t^n) - H(\mathbf{Y}_t^n)\right) - \left(H(Y_t, \mathbf{Y}_t^n, \mathbf{X}_{t-\Delta t}^n) - H(\mathbf{Y}_t^n, \mathbf{X}_{t-\Delta t}^n)\right) . \quad (4)$$

Intuitively, this tells us that if there is a causal relationship between X and Y with the correct direction and lag, then adding knowledge about Y_t brings more information to a system which does not know X, than to one which does (as X can partially predict it).

3.3. Statistical significance analysis

Once we know the transfer entropy between the motion of the camera and the object, we need to decide if this relationship is significant enough to make predictions of the object movement. Tests of statistical significance are preferred rather than comparing to a fixed threshold, since they offer theoretically founded decisions with probabilistic thresholds and explicitly cope with the inherent uncertainty caused by insufficient data.

To provide a sequence-specific baseline with no causal relationship the target time series Y is shuffled to remove any causality while retaining the distribution of amplitudes. We denote the shuffled signal as $\bar{Y}$. Then a Welch's t-test is performed, to obtain a p-value indicating the probability that both $T_{X\to Y}$ and $T_{X\to\bar{Y}}$ arose from the same distribution (a null hypothesis). When the observed causal relationship is statistically more significant than what is likely to arise by chance given the signal distributions, we conclude that it can be used for object motion prediction. This approach is shown to be successful in 15 out of 16 sequences from the VOT2013 dataset.

3.4. Finding the optimal parameters

When the causal relationship has been confirmed, we can attempt to predict the object motion from the overall movement of the whole scene (dual to the camera motion). Since different processes have, in general, different causal relationships, each particular sequence will have unique properties. In other words, we need to find an optimal set of parameters for subsequent prediction. These parameters are the time delay and the length of the time window, which can be seen as a mean and variance of the lag Δt. We want to pick these such that TE is maximal. Unfortunately, for the window length n this may not be as simple as in the case of Δt, as the transfer entropy stays high even when the window length is over-estimated (see Figure 4). We want to ensure that we do not miss any important information while using *only* important information. Excessively long time windows make the prediction unnecessarily slow without adding any significant gain. Also, non-discriminative features are likely to degrade performance, particularly with small training sets [14]. Therefore we define a relative improvement measure $f(\Delta t, n)$ (visualised by the column colours in Figure 4) from adding an additional frame to the window length and we require this relative improvement to be higher than a given threshold. The maximisation is then constrained as follows:

$$(\Delta t^*, n^*) = \arg\max_{\Delta t, n} T(\Delta t, n) \quad \text{s. t.} \quad f(\Delta t, n) > \theta_f , \quad (5)$$

with f defined as

$$f(\Delta t, n) = \frac{T(\Delta t, n) - \max\limits_{\overline{\Delta t}, \bar{n} < n} T(\overline{\Delta t}, \bar{n})}{\max\limits_{\overline{\Delta t}, \bar{n} \le n} T(\overline{\Delta t}, \bar{n})} , \quad (6)$$

where $T(\Delta t, n)$ relates to TE parameterised with a particular window length and lag. For experiments in this publication, $\theta_f = 10\,\%$ was used.

Figure 4 shows the effect of the parameters Δt and n. Notice the characteristic triangular shape of the area with consistently high TE: when we extend the time window, already containing the most relevant information, no significant information is gained or lost.

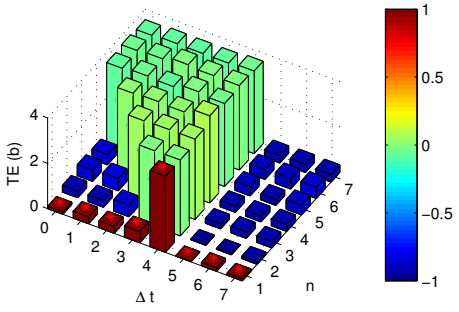


Figure 4. Dependence of TE on the time lag and the size of the time window. The column colours visualise the relative improvement f. The red stars denote all combinations of n and Δt with $f > \theta_f$.

4. Predicting the object motion

When using the video data, the signals are defined as follows. We assume having two multivariate time-series, $\mathbf{I}$ for the camera (image) and $\mathbf{O}$ for the object. We use multivariate signals: x and y coordinates and size (bounding box diagonal length), but additional dimensions (such as rotation) are possible. For the camera, the measured quantity is the image position relative to the first image. This is expressed in pixels, and is defined by the accumulated inter-frame motion $(\Delta x_t, \Delta y_t, \Delta s_t)^\top$ *i.e.* $\mathbf{I}_t = \left(\sum_{\tau=1}^t \Delta x_\tau, \sum_{\tau=1}^t \Delta y_\tau, s_0 \prod_{\tau=1}^t \frac{\Delta s_\tau}{s_\tau}\right)^\top$.

The global motion of the camera can be estimated robustly using the inter-frame shift of the whole image, with higher reliability than the object tracking. In our implementation we use a simple approach based on feature matching and RANSAC, but a more complicated method (e.g.

based on tracklets like in FoT [28]) may be used in challenging scenarios. Therefore, any discovered relationship can be used to transfer information from one (reliably estimated) signal to the other. In this publication, this information transfer is seen as the estimation of a distribution of possible poses for an object, based on its history $\mathbf{O}$ and additionally on its relation to the information from the image signal $\mathbf{I}$. This distribution can be supplied to a tracker as prior information to guide the tracking process. We examine two different approaches to object position prediction; the following sections describe these. For an intuitive comparison of both methods of prediction see Figures 5 and 6 (only x coordinate prediction is shown).

4.1. Window-based prediction

In the first case, a window-based prediction is used, similar to a non-linear autoregressive model. This approach is intuitively closer to the transfer-entropy background as described in Section 3.2. In autoregression, the current state is estimated (predicted) using a learned autoregressive function ϕ_a from its own history: $y_t = \phi_a(\mathbf{y}_t^n)$. Knowing there is a causal relationship between the two signals, the prediction can be improved using the other signal: $y_t = \phi_w(\mathbf{y}_t^n, \mathbf{x}_{t-\Delta t}^n)$, or more particularly:

$$\mathbf{O}_t = \phi_w(\mathbf{O}_t^{n^*}, \mathbf{I}_{t-\Delta t^*}^{n^*}) \,. \tag{7}$$

The window-based regressive function ϕ_w can be learned, taking a machine-learning approach. In other words, we take a set of all windows from the history and learn a regression (mapping) from the known part ($\mathbf{O}_t^n$ or both $(\mathbf{O}_t^{n^*}, \mathbf{I}_{t-\Delta t^*}^{n^*})$ knowing the optimal parameters) to the current pose $\mathbf{O}_t$. The principle of this is visualised in Figure 5.

4.2. Time-based prediction

In the second case, both the object position and the image position are modelled as functions of time $\mathbf{I}_t$ and $\mathbf{O}_t$. A sequential version of the autoregressive function can be learned, using the information about the data sequentiality: $\mathbf{O}_t = \phi_s(t \mid \mathbf{O}_{1..t-1})$.

Exploiting the causal knowledge, the $\mathbf{I}$ signal is shifted forward by the lag found as described in Section 3.4, Equation (5), to create $\mathbf{I}_{t-\Delta t^*}$ (aligning the signals). Then machine learning is used again, to learn the relationship between the two time-aligned signals, and to predict the future changes of $\mathbf{O}$. We again define a (time-based) regressive function ϕ_t such that

$$\mathbf{O}_t = \phi_t(t \mid \mathbf{O}_{1..t-1}, \mathbf{I}_{1..t-\Delta t^*}, n^*) \,. \tag{8}$$

The window length n^* is used as a measure of uncertainty in the timing of $\mathbf{I}$, *i.e.* how large a part of $\mathbf{I}$ is necessary to be taken into account during the prediction. In other words, both $\mathbf{I}$ and $\mathbf{O}$ are modelled as functions of time with the former guiding prediction of the latter in areas of insufficient data. The time-based function ϕ_t is visualised in Figure 6.

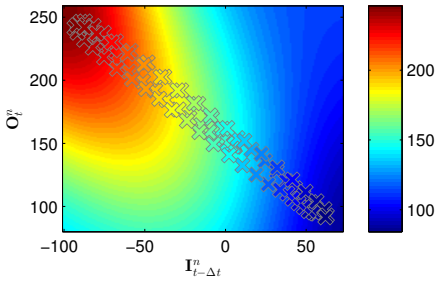


Figure 5. Window-based prediction function ϕ_w for the JUICE sequence. The training data are denoted by crosses, the background colour illustrates prediction of $\mathbf{O}_t$, given $\mathbf{O}_t^n$ and $\mathbf{I}_{t-\Delta t}^n$, by the learned function ϕ_w. In this case $n = 1$, however in general $\mathbf{O}_t^n$ and $\mathbf{I}_{t-\Delta t}^n$ can be high-dimensional.

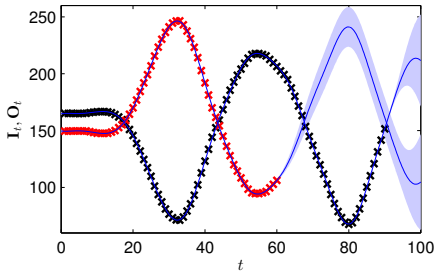


Figure 6. Time-based prediction with a learned relationship between the signals for the JUICE sequence ($\mathbf{I}$ is shifted for compactness). Black&red: the training data, the $\mathbf{I}$ and $\mathbf{O}$ signal, respectively; blue: mean and 95 % confidence intervals of the prediction. The learned relationship ensures prediction of $\mathbf{O}$ (by ϕ_t) for frames 60–90 has higher accuracy and confidence than would be possible with simple extrapolation.

5. Experimental evaluation

For our experiments, we implemented the proposed method as follows. On the signals $\mathbf{I}$ and $\mathbf{O}$ we perform the causality analysis, using $T_{\mathbf{I}\to\mathbf{O}}$; the $\mathbf{I}$ signal takes the role of X as used in Section 3.2 while $\mathbf{O}$ represents Y. For an initial coarse estimation of the signals lag, several overlapping windows with fixed length are used and TE with its statistical significance is computed for each of them in each frame, using Equation (4). When the statistical significance of any window exceeds a specified significance level α, a causal relationship is assumed and the optimal set of parameters found according to Equation (5). If no window is significant enough, we assume there is currently no causal relationship between the camera and the object motion. In our experiments, we used $n = 4$ and $\Delta t \in \{-4, -7, -10, \dots\}$, and a conservative significance level $\alpha = 0.01\,\%$.

The prediction was carried out as described in Sections 4.1 and 4.2. For the machine-learning stage, *Gaussian Process Regression* (GPR) was employed [32], as a probabilistic non-parametric regression approach, robust to overfitting. In all cases we used a combination of an RBF

and a bias kernel. Since different video-sequences in general do not share their causal properties, all the predictions were made using online sequence-specific learning. In other words, a tracker is required to track successfully for some time at the beginning of the sequence and this initialisation is used to learn the properties of the sequence (this implies that early tracking failures may lead to incorrect causal relationships, which would be however rejected as not significant). The prediction would be then supplied to the tracker as prior information and its tracking result would be added to the history data for a new prediction. For time efficiency, the ϕ-functions learning was initialised using ϕ from the previous frame.

For the window-based autoregression, windows containing a short history (3 frames: $\mathbf{O}_t^3$) of the position signal $\mathbf{O}$ were taken as features to predict the position in the consecutive frame $\mathbf{O}_t$ (to learn ϕ_a). Any other temporal information (inter-window relationships) were discarded, treating the data as a bag of equally important training inputs. The function ϕ_a was then learned and queried with the current history window $\mathbf{O}_t^3$ to obtain the prediction. The window-based causal prediction was done in a similar manner. The history windows $\mathbf{O}_t^{n^*}$ and $\mathbf{I}_{t-\Delta t^*}^{n^*}$ were concatenated together into $(3 \times 2 \times n^*)$-dimensional features ($3\times$ because of both $\mathbf{I}_t$ and $\mathbf{O}_t$ being $(x, y, s)^\top$ vectors), and the function ϕ_w was trained on the available history.

In the case of the time-based prediction via ϕ_s, the independent features are simply the frame indices and the dependent features the coordinates. For the causal prediction (the ϕ_t function), we need a technique to tie two signals together in an *a priori* unknown relationship. This can be achieved using a *coregionalisation* in the GPR. Coregionalisation is a technique which can model both signals $\mathbf{O}_t$ and $\mathbf{I}_{t-\Delta t*}$ as functions of time with a hidden relationship. Knowing the shape of one of the signals ($\mathbf{I}$) then guides the prediction of the other one ($\mathbf{O}$) even in locations distant from any training points of $\mathbf{O}$, as shown in Figure 6.

There are periods in the sequences, where no causal relationship was detected, and therefore no prediction parameters exist. In such places, the causal prediction is replaced by the appropriate autoregressive function: ϕ_w by ϕ_a and ϕ_t by ϕ_s. This explains the identical results for the sequences without causality during quantitative experiments in Section 5.3 (HAND, JUMP and TORUS).

5.1. Evaluation of causality detection

For the causality detection evaluation, the ICCV VOT Challenge 2013 [25] dataset was used (16 sequences, each containing between 172 and 770 frames). There are no "ground-truth causal relationships" we could use to measure the quality of our detection on the sequences (with the exception of zero relationship in case of static camera). However, the detected relationships are consistent with intuitive understanding of the scene dynamics and the optimal prediction parameters fulfilled our expectations. This shows that it is possible to use information-theory based measures to discover and quantify relationships between signals in real sequences for the task of visual object tracking. Using these, we can measure if there is a causal relationship between a camera and an object in a video-sequence, in which parts of the sequence, and we can measure its properties.

Sequence	Length	Length ratio	Δt^*	n^*
BICYCLE	271	81.2 %	-3	7
BOLT	350	64.3 %	-1	2
CAR	374	64.7 %	-10, -14	7, 5
CUP	303	53.5 %	-3, -3	4, 8
DAVID	770	94.9 %	-2, -1	2, 1
DIVING	231	40.3 %	-11	8
FACE	415	91.3 %	-1	1
GYMNASTICS	207	81.6 %	-1	1
HAND	244	0.0 %	NA	NA
ICESKATER	500	92.4 %	-11, -2	3, 7
JUICE	404	90.6 %	-1	1
JUMP	228	0.0 %	NA	NA
SINGER	351	82.3 %	-17, -13, -11	6, 1, 1
SUNSHADE	172	59.3 %	-8	8
TORUS	264	0.0 %	NA	NA
WOMAN	597	49.6 %	-8, -3, -3	5, 8, 1
Average	355	59.1 %		

Table 1. Causal detections on the VOT2013 dataset – detected durations and properties.

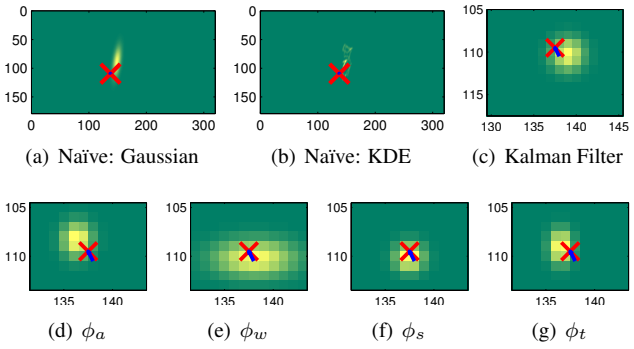
See Table 1 for the results. In the third column, we show the fraction of the sequence marked as containing a significant causal relationship. Then the optimal parameters for prediction are shown; in the case of different relationships for different time periods there are multiple parameter sets (*e.g.* 1 for BICYCLE, none for HAND or 3 for SINGER).

In the case of the sequence JUMP, none of the detected causal relationships were statistically significant. However, this is not necessarily an error as although the camera is not static, we do not know if a true relationship exists between object and camera motion. There are three sequences with a static camera (constant zero $\mathbf{I}$) and therefore no causal relationships, these are marked in grey in the tables. For two of them, this was correctly detected using TE. For the CAR sequence, a causal relationship was incorrectly discovered due to inaccurate estimation of $\mathbf{I}$. However, this means that causality detection only failed in 1 out of 16 sequences. It is also worth pointing out the relatively common occurrence of the (-1,1) pair, indicating the immediate causal effect of a moving camera on the apparent motion of a static object.

See the supplementary material for causality detection evaluation on synthetic data, where the GT is known.

5.2. Qualitative prediction evaluation

The task we are given during the prediction stage is to estimate the distribution of the possible object positions to be supplied to the tracker as a prior. For a performance measure, there is a requirement to discover how well the ground truth position (GT) is represented by this distribution. While simple distance between the distribution mean



(a) Naïve: Gaussian (b) Naïve: KDE (c) Kalman Filter

(d) ϕ_a (e) ϕ_w (f) ϕ_s (g) ϕ_t

Figure 7. Qualitative prediction results on the DIVING sequence, frame #200. Predicted distributions shown with the ground truth position (in red) and ground truth inter-frame shift (blue) overlaid. See the text for discussion.

and the GT indicates the prediction accuracy, it does not take uncertainty into account. In particular, if two predictors predict the same correct position, the one with high confidence is of most benefit to the tracker. This holds in the opposite direction as well, for a misprediction it is better to report lower certainty. For these reasons, we used the probability density function (PDF) as a performance measure in our experiments. Additionally, we have integrated the error over the entire PDF support region to obtain expectation for the prediction error. In the following section, we use the mean values of both across each sequence as quantitative measures.

The causality-based prediction was compared with several alternative approaches as follows. Firstly, two naïve approaches are examined, treating all historical states as equally important, based on the assumption that the object stays in an approximately stable location. One models the distribution as a Gaussian, as used in the re-detector of Lebeda *et al*. [26], while the other one uses a Kernel Density Estimation (KDE) to model the PDF with higher accuracy. These can be seen as implementation of the central bias and stable location priors. Visual tracking algorithms often use a *Kalman filter* (KF), or its extension, as their internal motion model [8, 22, 24, 40]. Therefore, the KF is a natural alternative to causal prediction. Finally, we evaluated the autoregressive functions ϕ_a and ϕ_s, and the causal predictions ϕ_w and ϕ_t.

The results are visualised in Figure 7. In both cases, the GT is well inside the naïvely predicted distributions. However, these distributions are spread over the whole image and therefore the PDF is relatively low. In the case of the KF, the predictions lag behind the true signal somewhat, causing mispredictions. The autoregression given by ϕ_a and ϕ_s helps significantly with the GT being at least at the edge of the predicted distribution; ϕ_s shows better performance than ϕ_a. Window-based causal prediction gives accurate modes for the distribution, although the long window in the case of DIVING ($n^* = 8$) results in a low confidence and thus a lower PDF. The time-based prediction performed the best of all the tested predictors; ϕ_t predicted positions close to the GT while having an appropriate confidence.

5.3. Quantitative prediction evaluation

As previously mentioned, the mean PDF and mean expectation of error across each sequence were used as performance measures in the quantitative evaluation. See Table 2 for the results. In the CAR sequence, the tracked car stops for a large part of the sequence in one location, the KDE predicted very high probability for this location, which is reflected by a very high mean PDF. A similar phenomenon can be found in GYMNASTICS, with the tracked person standing in one place for a part of the sequence.

In the BOLT and ICESKATER sequences, the $\mathbf{I}$ signal estimation failed for one region of each sequence due to very low texture of the background. This renders the relationship between the camera and object motions unstable and therefore the ϕ_w and ϕ_t predictors have lower performance in these sequences. This is more noticeable in the case of error expectation, where these outliers render the ϕ_t prediction to not have the lowest average error, despite being the lowest on majority of sequences.

In general, disregarding these outliers, several statements can be made about the performance of the compared predictors. Both global probability distributions have an image-wide spread and therefore a very low PDF. Prediction using KF is better localised and has therefore significantly better performance, although still worse than the learned regressive functions. For the learned functions, we can say that time-based ones (ϕ_s and ϕ_t) in general perform better than window-based ϕ_a and ϕ_w. Regressive function ϕ_w performs slightly worse than its non-causal counterpart ϕ_a, due to the lower confidence of the prediction (higher variance and therefore lower PDF). The time-based causal function ϕ_t was shown as the best predictor, beating the second best by a large margin (62 %). In addition, it performs more than three times better than KF, which is a commonly used motion model.

Table 3 show the effect of causal prediction on the performance of the state of the art trackers FoT [28] and Struck [17]. The tabulated values are the VOT accuracy/robustness metrics [25] – mean bounding box overlap (higher is better) and number of failures per sequence (lower is better). We compare against vanilla trackers and a simple background motion compensation (BMC), using the image context but no temporal causal relationships. While the simple camera motion information does not prove useful, supplying the tracker prior information from causality-based prediction improves its performance significantly. In general, robustness is affected only slightly, while the main improvement is in the accuracy domain. For

Sequence	Expectation of error (px)							Mean probability density (-)						
	Gaussian	KDE	KF	ϕ_a	ϕ_w	ϕ_s	ϕ_t	Gaussian	KDE	KF	ϕ_a	ϕ_w	ϕ_s	ϕ_t
BICYCLE	22.6	22.8	3.8	3.4	19.3	3.2	**3.1**	0.001	0.002	0.025	0.028	0.010	0.031	**0.036**
BOLT	64.1	64.7	**3.5**	3.9	63.5	4.2	6.8	0.001	0.004	**0.030**	0.022	0.013	0.020	0.015
CAR	52.2	64.6	2.3	1.5	1.5	1.4	**1.0**	0.006	**0.577**	0.060	0.123	0.124	0.130	0.524
CUP	22.5	22.8	3.5	1.8	2.2	1.7	**1.5**	0.012	0.015	0.022	0.086	0.065	0.093	**0.121**
DAVID	25.4	25.3	**5.2**	5.8	5.5	5.4	5.3	0.001	0.001	**0.013**	0.009	0.009	0.010	0.011
DIVING	18.7	19.1	2.5	1.8	1.9	1.8	**1.7**	0.007	0.009	0.040	0.085	0.081	0.089	**0.096**
FACE	16.3	4.6	2.0	1.5	1.6	1.4	**1.3**	0.002	0.013	0.068	0.118	0.108	0.124	**0.206**
GYMNASTICS	18.2	22.6	4.2	3.5	12.5	**2.5**	3.7	0.021	**0.193**	0.044	0.069	0.045	0.070	0.109
HAND	81.0	80.6	9.0	**5.6**	**5.6**	6.1	6.1	0.002	0.002	0.001	**0.010**	**0.010**	**0.010**	**0.010**
ICESKATER	31.2	31.1	3.9	**2.5**	648.4	2.7	14.3	0.001	0.001	0.022	**0.061**	0.006	0.053	0.008
JUICE	71.7	72.0	10.7	3.1	2.1	2.5	**2.0**	0.006	0.009	0.001	0.039	0.083	0.045	**0.088**
JUMP	40.2	38.6	2.5	**1.7**	**1.7**	1.8	1.8	0.001	0.003	0.041	**0.102**	**0.102**	0.096	0.096
SINGER	77.2	69.8	**3.2**	3.6	20.1	3.6	5.4	0.000	0.001	**0.030**	0.022	0.010	0.021	0.014
SUNSHADE	56.4	56.3	10.1	9.8	41.4	4.5	**3.9**	0.000	0.001	0.003	0.012	0.005	0.018	**0.021**
TORUS	63.2	62.7	6.0	3.3	3.3	**3.1**	**3.1**	0.004	0.003	0.007	**0.031**	**0.031**	0.026	0.026
WOMAN	34.8	35.2	3.9	3.4	6.3	3.0	**2.9**	0.000	0.001	0.027	0.059	0.046	0.060	**0.069**
Average	43.5	43.3	4.8	3.5	52.3	**3.1**	4.0	0.004	0.052	0.027	0.055	0.047	0.056	**0.091**

Table 2. Quantitative results of the prediction on the VOT2013 dataset. The best and second best results are denoted by a bold typeface and underlining respectively (separately for error expectation and mean PDF; multiple columns highlighted in cases of equal values).

Sequence	FoT	FoT_{BMC}	FoT_{ϕ_t}	Struck	$\text{Struck}_{\text{BMC}}$	Struck_{ϕ_t}
BICYCLE	0.70/ **1**	0.70/ **1**	**0.71/ 1**	0.43/0.3	0.39/0.2	**0.54/0.0**
BOLT	0.46/14	**0.59/13**	0.52/**13**	**0.76/3.7**	0.58/8.5	0.72/5.4
CAR	0.55/ **1**	0.53/ **1**	**0.59/ 1**	0.40/**0.0**	**0.42/0.0**	0.38/**0.0**
CUP	0.81/ **0**	0.80/ **0**	**0.82/ 0**	0.78/**0.0**	**0.83/0.0**	0.82/**0.0**
DAVID	**0.76/ 0**	0.59/ **0**	0.75/ **0**	0.67/0.7	0.60/**0.5**	**0.70**/0.9
DIVING	0.25/ 5	**0.32/ 3**	0.25/ 5	**0.39/1.0**	0.36/**1.0**	0.36/**1.0**
FACE	0.74/ **0**	**0.84/ 0**	0.78/ 1	**0.83/0.0**	0.80/**0.0**	**0.83/0.0**
GYMNASTICS	**0.63**/ 6	0.60/ **4**	0.61/ 6	0.55/**2.3**	**0.59**/3.9	0.56/4.0
HAND	**0.40**/ 4	0.38/ **3**	0.38/ 4	**0.52/4.1**	**0.52**/4.6	**0.52/4.1**
ICESKATER	0.43/10	**0.45**/10	0.38/ **4**	**0.62/0.0**	0.32/9.4	0.54/0.7
JUICE	0.88/ **0**	**0.93/ 0**	0.90/ **0**	0.65/**0.0**	**0.91/0.0**	0.89/**0.0**
JUMP	0.62/ 1	0.71/ **0**	**0.72/ 0**	0.56/**0.0**	**0.57/0.0**	**0.57/0.0**
SINGER	**0.74/ 0**	0.65/ **0**	**0.74/ 0**	0.30/**0.0**	**0.41**/1.0	0.33/**0.0**
SUNSHADE	0.59/ 2	0.57/ **1**	**0.76/** 2	**0.77/0.0**	**0.77/0.0**	0.74/**0.0**
TORUS	0.73/ **0**	**0.75/** 1	0.72/ **0**	0.49/**4.3**	0.55/5.2	**0.56**/5.1
WOMAN	0.61/ **0**	0.12/ 1	**0.71/** 5	**0.75/0.0**	0.65/**0.0**	0.74/**0.0**
Average	0.62/2.8	0.60/**2.4**	**0.65**/2.6	0.59/**1.0**	0.58/2.1	**0.61**/1.3

Table 3. Tracking results on the VOT2013 benchmark. The best and second best results are highlighted separately for the FoT/Struck families of trackers and for accuracy/robustness.

Category	ASLA[21]	SCM[41]	Struck[17]	Struck	$\text{Struck}_{\text{BMC}}$	Struck_{ϕ_t}
BC	0.59/3.0	**0.61**/2.9	0.59/3.3	0.60/1.9	0.55/1.9	**0.61/1.7**
DEF	0.51/4.5	0.52/4.8	0.52/4.6	0.55/2.4	0.55/2.6	**0.60/2.2**
FM	0.42/6.5	0.43/6.5	0.56/3.8	0.53/3.2	0.51/3.3	**0.57/2.5**
IPR	0.52/4.1	0.52/4.3	**0.57**/3.4	0.53/2.6	0.50/3.0	0.55/**2.0**
IV	0.60/3.0	**0.61**/3.1	0.59/3.3	0.58/2.1	0.51/1.9	0.60/**1.6**
LR	0.59/2.3	**0.62**/2.5	0.59/3.9	0.51/1.4	0.48/1.1	0.56/**1.0**
MB	0.45/5.9	0.45/5.9	**0.60**/2.8	0.53/3.0	0.51/2.9	0.56/**2.0**
OCC	0.56/3.8	0.57/3.8	0.56/4.1	0.55/2.5	0.54/2.9	**0.59/2.0**
OPR	0.56/3.7	0.57/3.8	0.57/3.7	0.55/2.3	0.54/2.7	**0.59/1.9**
OV	0.55/4.3	0.56/4.5	**0.59**/3.4	0.55/3.0	0.58/2.7	0.55/**2.5**
SV	0.54/3.9	0.56/3.9	**0.58**/3.6	0.52/2.3	0.49/2.7	0.57/**1.9**
All	0.53/4.1	0.54/4.1	0.57/3.6	0.55/2.4	0.51/2.6	**0.59/1.9**

Table 4. Tracking results on VTB1.1.[2] Results in the last three columns were obtained using the VOT evaluation criteria, using the VTB criteria would improve the accuracy even further.

FoT and the ICESKATER sequence, there is a marginal drop in accuracy, which is more than balanced by a dramatic increase in robustness, lowering the number of failures by 60 %. For comparison, we have carried out the same experiments with the zero-order tracker. While it works in some cases, the mean performance is significantly poorer: accuracy of 0.34 and robustness 6.25.

Additionally, we have carried out experiments on the much larger Visual Tracking Benchmark (VTB1.1 [37]). The Struck tracker is currently at the head of the leaderboard. As shown in Table 4, using our causal predictions further improves this — by more than the current difference between the first two trackers — leading to a new state-of-the-art on this benchmark.

6. Summary

In this paper, we have explored causal relationships between object and camera motions. We have proposed an approach to discover and quantify this relationship using transfer entropy, a statistical tool which to our knowledge has not been used in any previous publication in the area of computer vision. We have also shown that it is possible to find the optimal time window for prediction of the object position based on the global image motion even for complex non-linear relationships. Finally, these causality-based motion predictions were evaluated on a range of standard tracking sequences, and shown to offer excellent performance (increasing average prediction accuracy by 62 % and improving the top performing tracker on VTB1.1 by 7 % in accuracy and 22 % in robustness), with particular robustness to camera shakes and fast motion. These are typically the greatest source of errors in modern tracking, as shown in the recent VOT challenge, and thus the proposed techniques, which we will make publicly available, provide an invaluable addition to any tracking algorithm.

Acknowledgement: This work was supported by the EPSRC project EP/I011811/1: Learning to Recognise Dynamic Visual Content from Broadcast Footage, and the Rabin Ezra Scholarship.

[2] BC: Background Clutter, DEF: Deformation, FM: Fast Motion, IPR: In-Plane Rotation, IV: Illumination Variation, LR: Low Resolution, MB: Motion Blur, OCC: Occlusion, OPR: Out-of-Plane Rotation, OV: Out-of-View, SV: Scale Variation.

References

[1] Aristotle. *Physics II.* 2

[2] B. Babenko, M.-H. Yang, and S. Belongie. Visual tracking with online multiple instance learning. *PAMI*, 2011. 3

[3] M. Bauer, J. Cox, M. Caveness, J. J. Downs, and N. Thornhill. Finding the direction of disturbance propagation in a chemical process using transfer entropy. *Control Systems Technology*, 2007. 2

[4] M. Brand. Physics-based visual understanding. *CVIU*, 1996. 2

[5] L. Cehovin, M. Kristan, and A. Leonardis. Robust visual tracking using an adaptive coupled-layer visual model. *PAMI*, 2013. 3

[6] N. Dowson and R. Bowden. A unifying framework for mutual information methods for use in non-linear optimisation. In *ECCV*, 2006. 3

[7] N. Dowson, T. Kadir, and R. Bowden. Estimating the joint statistics of images using nonparametric windows with application to registration using mutual information. *PAMI*, 2008. 3

[8] G. A. Einicke and L. B. White. Robust extended Kalman filtering. *Signal Processing*, 1999. 7

[9] Y. Fan, H. Yang, S. Zheng, H. Su, and S. Wu. Video sensor-based complex scene analysis with Granger causality. *Sensors*, 2013. 2

[10] A. Fire and S.-C. Zhu. Using causal induction in humans to learn and infer causality from video. In *Annual Conf. of the Cognitive Science Society*, 2013. 2

[11] K. Friston, R. Moran, and A. K. Seth. Analysing connectivity with Granger causality and dynamic causal modelling. *Current Opinion in Neurobiology*, 2013. 2

[12] H. Grabner, C. Leistner, and H. Bischof. Semi-supervised on-line boosting for robust tracking. In *ECCV*, 2008. 3

[13] C. W. J. Granger. Investigating causal relations by econometric models and cross-spectral methods. *Econometrica*, 1969. 2

[14] I. Guyon and A. Elisseeff. An introduction to variable and feature selection. *JMLR*, 2003. 4

[15] R. S. Hacker and A. Hatemi-J. Tests for causality between integrated variables using asymptotic and bootstrap distributions: theory and application. *Applied Economics*, 2006. 2

[16] J. P. Hamilton, G. Chen, M. E. Thomason, M. E. Schwartz, and I. H. Gotlib. Investigating neural primacy in Major Depressive Disorder: multivariate Granger causality analysis of resting-state fMRI time-series data. *Molecular Psychiatry*, 2011. 2

[17] S. Hare, A. Saffari, and P. H. S. Torr. Struck: Structured output tracking with kernels. In *ICCV*, 2011. 2, 3, 7, 8

[18] C. Hiemstra and J. D. Jones. Testing for linear and nonlinear Granger causality in the stock price-volume relation. *J. of Finance*, 1994. 2

[19] K. Hlavackova-Schindler, M. Palus, M. Vejmelka, and J. Bhattacharya. Causality detection based on information-theoretic approaches in time series analysis. *Physics Reports*, 2007. 3

[20] D. Hume. *A Treatise of Human Nature*. 1738. 2

[21] X. Jia, H. Lu, and M.-H. Yang. Visual tracking via adaptive structural local sparse appearance model. In *CVPR*, 2012. 8

[22] S. Julier and J. Uhlmann. New extension of the Kalman filter to nonlinear systems. In *Signal Processing, Sensor Fusion and Target Recognition*, 1997. 7

[23] A. Kaiser and T. Schreiber. Information transfer in continuous processes. *Physica D: NP*, 2002. 2

[24] R. E. Kalman. A new approach to linear filtering and prediction problems. *J. of Basic Engineering*, 1960. 7

[25] M. Kristan, R. Pflugfelder, A. Leonardis, J. Matas, F. Porikli, et al. The visual object tracking VOT2013 challenge results. In *ICCV VOT*, 2013. 1, 6, 7

[26] K. Lebeda, S. Hadfield, J. Matas, and R. Bowden. Long-term tracking through failure cases. In *ICCV VOT*, 2013. 3, 7

[27] J. Matas. Visual tracking in the 21st century. In *BMVC*, 2012. 1, 2

[28] J. Matas and T. Vojir. Robustifying the flock of trackers. In *CVWW*, 2011. 2, 5, 7

[29] S. Narayan and K. Ramakrishnan. A cause and effect analysis of motion trajectories for modeling actions. In *CVPR*, 2014. 2

[30] J. Pearl. *Causality: Models, Reasoning, and Inference*. Cambridge University Press, 2000. 2

[31] K. Prabhakar, S. Oh, P. Wang, G. Abowd, and J. Rehg. Temporal causality for the analysis of visual events. In *CVPR*, 2010. 2

[32] C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. MIT Press, 2006. 5

[33] T. Schreiber. Measuring information transfer. *Physical Review Letters*, 2000. 2

[34] H. Sumioka, Y. Yoshikawa, and M. Asada. Learning of joint attention from detecting causality based on transfer entropy. *J. of Robotics and Mechatronics*, 2008. 2

[35] D. L. Thornton and D. S. Batten. Lag-length selection and tests of Granger causality between money and income. *J. of Money, Credit and Banking*, 1985. 2

[36] R. Vicente, M. Wibral, M. Lindner, and G. Pipa. Transfer entropy – a model-free measure of effective connectivity for the neurosciences. *J. of Comp. Neurosc.*, 2011. 2

[37] Y. Wu, J. Lim, and M.-H. Yang. Online object tracking: A benchmark. In *CVPR*, 2013. 8

[38] J. Xiao, R. Stolkin, and A. Leonardis. An enhanced adaptive coupled-layer LGTracker++. In *ICCV VOT*, 2013. 3

[39] S. Yi and V. Pavlovic. Sparse Granger causality graphs for human action classification. In *ICPR*, 2012. 2

[40] A. Yilmaz, O. Javed, and M. Shah. Object tracking: A survey. *ACM Computing Surveys*, 2006. 7

[41] W. Zhong, H. Lu, and M.-H. Yang. Robust object tracking via sparsity-based collaborative model. In *CVPR*, 2012. 8