Cross-Domain Traffic Scene Understanding by Motion Model Transfer

Xun Xu
Queen Mary, University of London
London E1 4NS, UK
xx302@eecs.qmul.ac.uk

Shaogang Gong
Queen Mary, University of London
London E1 4NS, UK
sgg@eecs.qmul.ac.uk

Timothy Hospedales
Queen Mary, University of London
London E1 4NS, UK
tmh@eecs.qmul.ac.uk

ABSTRACT

This paper proposes a novel framework for cross-domain traffic scene understanding. Existing learning-based outdoor wide-area scene interpretation models suffer from requiring long term data collection in order to acquire statistically sufficient model training samples for every new scene. This makes installation costly, prevents models from being easily relocated, and from being used in UAVs with continuously changing scenes. In contrast, our method adopts a geometrical matching approach to relate motion models learned from a database of source scenes (source domains) with a handful sparsely observed data in a new target scene (target domain). This framework is capable of online "sparse-shot" anomaly detection and motion event classification in the unseen target domain, without the need for extensive data collection, labelling and offline model training for each new target domain. That is, trained models in different source domains can be deployed to a new target domain with only a few unlabelled observations and without any training in the new target domain. Crucially, to provide cross-domain interpretation without risk of dramatic negative transfer, we introduce and formulate a scene association criterion to quantify transferability of motion models from one scene to another. Extensive experiments show the effectiveness of the proposed framework for cross-domain motion event classification, anomaly detection and scene association.

Categories and Subject Descriptors

I.2.10 [**Artificial Intelligence**]: Vision and Scene Understanding—*Motion, Representations, data structures, and transforms*; I.4.7 [**Image Processing And Computer Vision**]: Feature Measurement—*Invariants*

Keywords

Transfer Learning; Anomaly Detection; Visual Surveillance; Gaussian Mixtures

ARTEMIS'13, October 21, 2013, Barcelona, Spain.

http://dx.doi.org/10.1145/2510650.2510657.

1. INTRODUCTION

With the proliferation of video surveillance systems, unusual behaviour and event detection systems are in increasing demand. Thousands of existing surveillance cameras are producing gigantic amounts of surveillance data. Automatic interpretation of unusual events of interest in wide-area public scenes such as road traffic is highly desirable. Many studies have thus focused on automatic learning of traffic scene models, with which anomalies can be detected [10, 5, 23, 8]. However, a readily re-deployable model trained from one scene for applying to another still does not exist because current approaches rely on two strong assumptions: (1) long term data collection and learning in a target domain is always possible, and/or (2) the target and training domains have the same or very similar characteristics in both their motion event distribution and feature space representation. As a result, most models require training and deployment on the same scene, and cannot be applied readily across-domain [6, 14]. Consequently for each new scene deployment, one needs to collect new data and retrain the model from scratch. For supervised classification of behaviours, this requires costly annotation of new data. For scenes with sparse activities, collecting a statistically sufficient volume of training data may take a prohibitive amount of time and manpower. In recent years, the increasing need for non-stationary Unmanned Aerial Vehicle (UAV) surveillance has further highlighted the limitations of model construction based on these two strong assumptions. For typical UAV scenarios, event classification and anomaly detection from aerial data sources require an immediate response and preclude the possibility of collecting enough training data for a scene, let alone annotating data and re-learning models for continuously changing scenes. Therefore, cross-domain scene interpretation has increasingly become not only desirable but also critical for practical deployment and non-stationary surveillance.

To address this problem, in this work we introduce a novel cross-domain scene interpretation framework. This framework performs offline learning on a batch of multiple source domains and then provides online interpretation of object activity events in an unseen target domain by spatio-temporal matching to the source domain observations. In this way, we can leverage an arbitrary (and increasing) volume of prior data, annotation and learned models in order to allow a new target domain scenario to be interpreted with only few bootstrapping observations and no supervision. We call this cross-domain "sparse-shot" event classification and anomaly detection. The sparse unlabelled observations in

the target domain are utilised online to enable automatic selection of the most relevant source domains for model transfer. Specifically, we use tracking data of object motion in each domain and compare tracks across-domain using the Kullback-Leibler Divergence (KLD) between their mixture of Gaussians (GMM) representations. To compare tracks across heterogeneous domains, we optimise KLD over similarity transforms, thus achieving a similarity-transform/domain invariant distance metric. Building on this capability, the proposed framework is able to compare/quantify entire domains for cross-domain similarity and thus match domains without negative scene knowledge transfer, and hence classify individual target-domain object tracks and quantify their abnormality. Crucially, this is all achieved without the typically required extensive target domain data collection, offline labelling and model training process.

1.1 Related Work

Currently, there are two broad categories of approaches to anomalous motion event detection depending on the features used [19]. One category of approaches employs low-level image features [10, 5, 23], while the other category generates high-level features by performing target detection and tracking [6, 14, 8]. Low-level feature approaches employ, for example, background subtraction [10], tracklets [23], granular particles [12] or optical flow [18, 21, 5]. Then motion events or activity patterns are learned from these features. The low-level feature approaches may be superior when the scene is extremely crowded or partially occluded so that tracking may fail to produce meaningful results. However, since individual objects are not identified in most of these approaches, detected anomalies often cannot be attributed to specific objects.

The second group of approaches extract motion tracks of individual objects by multi-target tracking [6, 13, 8]. Track based methods have also been extensively studied for anomaly detection, and have the advantage that a detected unusual event can be unambiguously associated to a particular object. Motion pattern or path models are usually learned beforehand from large volumes of training data, which may be clustered into particular templates of motion events such as "turn-right" or "go-straight". New trajectories can be classified by matching against the event-templates, or evaluated for abnormality by the likelihood under the entire model [8], or the distance to the nearest template [14].

Although good performance has been achieved in a variety of datasets using these techniques, a severe drawback of existing approaches is their highly problem/domain specific nature. Both low-level feature and track based methods require long-term collection of training data to be effective. For example, enough typical events (atomic events or trajectories) must be obtained to properly model the distribution of normal activities in order to perform accurate anomaly detection [8, 14, 5]. To overcome this problem, a recent study [7] proposed geometrical transformation to assist cross-domain motion pattern recognition. However, this model uses low-level optical flow features [18] and lacks of an explicit object model. To be able to associate events with individual objects is of great interest in anomaly detection. In our work we overcome this problem by analysing track features in a similarity transform invariant way, building on and leveraging the geometrical transform optimisation model.

Overall, our contributions are as follows: (1) Introduce and solve a novel problem of cross-domain sparse-shot abnormal motion event detection without any model training (supervised or unsupervised) in the target domain, (2) solve the problem of cross-domain sparse-shot object motion event (trajectory-based) classification and, (3) formulate a model to quantify cross-domain scene association relevance using the sparse target domain observations. This enables our framework to automatically match a target domain to its most similar/relevant source domain for motion model transfer and thus maximise cross-domain event recognition and anomaly detection while avoiding negative transfer.

2. METHODOLOGY

In this work we address sparse-shot object motion event classification and anomaly detection. To achieve this we first introduce a probabilistic model for object tracks (motion events) and track clusters (motion event models) (Section 2.1), followed by a view invariant distance metric for track matching (Section 2.2). Crucially, we also show how to quantify cross-domain transferability and select appropriate source domains for motion model transfer (Section 2.3). Finally, building on these two capabilities, we present a model for cross-domain sparse-shot anomaly detection and motion event classification (Section 2.4).

2.1 Motion Model and Event Representation

We first describe the construction of a probabilistic motion event and motion model representation from observed object motion tracks. We track each individual object (e.g., a vehicle in a traffic scene) using the tracker [17]. Each trajectory T_i of length n_i is represented by a sequence of coordinates and time-stamps $T_i = \{(x_{i,j}, y_{i,j}, t_{i,j})\}_{j=1...n_i}$. Note that this representation keeps the directional information via time-stamp t. We filter broken trajectories with a threshold on minimum length.

Modeling motion patterns.

To learn a domain-specific object motion event model in a given scene, we cluster object trajectories to obtain typical motion patterns or events for the scene. Before clustering, we normalise trajectory length. To that end, we employ the Douglas-Peucker algorithm [1] to segment the trajectory at a set of control points. A re-sampled trajectory with a fixed number of N_t points is then obtained by linearly interpolating each interval between proximal control points. Given this pre-processed set of length-normalised object motion trajectories, we over-cluster them using Fuzzy C means (FCM) with Euclidean distance [13] into a large number of C_0 clusters to ensure all modes of object behaviour in the given scene are represented. For each cluster $c = 1, \ldots, C_0$, we fit a N component Gaussian mixture model (GMM) to the fixed length trajectories in that cluster. Each trajectory cluster therefore has a probabilistic representation as $g_c(\mathbf{x}) = \sum_{k=1}^{K} w_k \mathcal{N}(\mathbf{x}; \mu_{k,c}, \Sigma_{k,c})$ (where μ and Σ are the mean and covariance of each GMM component). To reduce the computational burden, we eliminate redundant clusters from the over-clustering FCM step by computing pairwise KLD (see Section 2.2) between each cluster, and applying self-tuning spectral clustering [22] to determine automatically the optimal number C of motion patterns (trajectory clusters) representing the typical motion events in the

scene. This over-clustering followed by pruning process ensures that all the modes of variability in the scene are represented. Without this, direct clustering can result in the most common trajectory types dominating and less-common motion events not being modelled.

Modeling individual trajectories.

Finally, we need to establish a probabilistic representation of an individual run-time object trajectory to interpret under the motion patterns defined above. We define a GMM for each test trajectory by a Gaussian centred on each observation $[x_j, y_j, t_j]$ with diagonal covariance. x and y variance are set to the bounding-box size – since the object centre is somewhere in the bounding box – and t variance set to σ_t so to reflect maximum expected speed.

2.2 Cross-Domain Transfer of Motion Models

Given the proposed probabilistic representation of object motion events based on individual trajectories, we now describe a similarity measure to compare object motion models and events. We first describe the within-domain case before generalising to the across-domain case which needs to account for the potentially different scene geometries.

Within-domain comparisons.

The probabilistic (GMM) representation established for motion events and motion patterns has the advantage of modelling the covariance in motion patterns, however this necessitates some care in defining suitable similarity measures. To quantify the similarity between probabilistically represented motion events, we exploit the Kullback-Leibler Divergence [7] (KLD) which measures the similarity between distributions. Note that both object tracks and track clusters are modelled as distributions. The KLD between two distributions $g_m(\mathbf{x})$ and $g_t(\mathbf{x})$ is:

$$\mathcal{KLD}(g_m \parallel g_t) = \int g_m(\mathbf{x}) \log \left(\frac{g_m(\mathbf{x})}{g_t(\mathbf{x})} \right) \mathrm{d}\mathbf{x}. \qquad (1)$$

Since there is no analytical solution for the KLD in the case of GMMs distributions, we employ a Monte Carlo approximation [4]. M vectors $\mathbf{x}_m = [x_j, y_j, t_j]$ are sampled from $g_m(\mathbf{x})$ and used to approximate the KLD by evaluating their likelihood under $g_t(\mathbf{x})$:

$$\mathcal{KLD}(g_m \parallel g_t) \approx \frac{1}{M} \sum_{j=1}^{M} \log \left(\frac{g_m(\mathbf{x}_j^m)}{g_t(\mathbf{x}_j^m)} \right). \qquad (2)$$

Since a trajectory cluster typically has larger variance than an individual trajectory, the forward KLD is usually much greater than the backward KLD. To obtain a more stable similarity metric suitable for comparing both trajectories and clusters, we utilise the average of forward and backward measures ($\mathcal{KLD}(g_m \parallel g_t)$ and $\mathcal{KLD}(g_t \parallel g_m)$), i.e. the Jensen-Shannon Divergence (JSD) [11].

Cross-domain mapping.

We are ultimately interested in cross-domain motion model and motion event comparisons by mapping object motion trajectories and their clusters in order to facilitate across-domain scene understanding. For wide area surveillance, semantically equivalent motion events differing only in their view geometry can be arbitrarily different in image plane, but are equivalent under an geometric similarity transformation $\mathbf{H}$ (a 3×3 matrix). That is, the same, or two semantically equivalent scenes viewed from differing angles cannot be compared directly unless the translation, scaling and rotation ($\mathbf{H}$) that relates them is known. Therefore to define a distance measure capable of comparing trajectories and motion patterns across different view of unknown geometry, it should be invariant to the similarity transform $\mathbf{H}$ relating them.

To quantify the distance $D(g_m, g_t) = \mathcal{KLD}(g_m||g_t)$ (Eq. (2)) between GMM representations of trajectories and motion patterns in a view-independent way, we therefore optimize the metric $D^{\mathbf{H}}(g_m, g_t) = \mathcal{KLD}(g_m||g_t^{\mathbf{H}})$ for transformation $\mathbf{H}$ in each comparison. Here $g_t^{\mathbf{H}}$ indicates geometric transformation of the motion model $g_t(\mathbf{x})$ by $\mathbf{H}$. The optimal transformation $\mathbf{H}^*$ is the one that maximizes their similarity. For GMMs models under the distance approximation Eq. (2), this corresponds to maximising the likelihood of points sampled from GMM g_m under *transformed* GMM distribution $g_t^{\mathbf{H}}$:

$$\mathbf{H}^* = \underset{\mathbf{H}}{\operatorname{argmax}} \log \prod_{j=1}^{M} \sum_{k=1}^{K} w_k \mathcal{N}(\mathbf{x}_j; \mathbf{H}\mu_k, \mathbf{H}\Sigma_k\mathbf{H}^T). \qquad (3)$$

As in [7], we approximately optimise Eq. (3) by proxy of alternating estimating point correspondences using the Hungarian algorithm [9], and directly fitting $\mathbf{H}$ given fixed correspondences [3] (illustrated in Figure 1). This is necessary because without correspondence least-squares transformation estimation (LSE) is meaningless, but correspondence cannot be estimated unless the two patterns are in alignment.

In contrast to [7], there are three notable differences: (i) We do not need the path-context information required to avoid the local minima problem in [7]. This is because the local minima is in fact a mismatch in temporal order, whereas in our case the temporal information is already modelled by the third (time) dimension of our probabilistic trajectory model; (ii) We use the Hungarian-LSE alternation of [7] to get an initial condition, followed by direct optimisation for Eq. (3) using BFGS [15]. This is a better solution than solely optimising Eq. (3) by proxy [7], because there is no formal relation between the alternation and Eq. (3). Finally, (iii) for our final distance metric between trajectories or clusters m and t; both within and across scenes, we use the JSD (Eq. (4)) which allows us to make stable comparison between tracks and clusters whilst the model of [7] only performs cluster-cluster comparison.

$$D^{\mathbf{H}^*}(g_m, g_t) = \mathcal{KLD}(g_m||g_t^{\mathbf{H}^*}) + \mathcal{KLD}(g_t^{\mathbf{H}^*}||g_m). \qquad (4)$$

The methods described in this section enable cross-domain (similarity transform invariant) comparison of motion events (tracks and clusters). However the central issue with exploiting this capability in practice is that these comparisons are only useful / meaningful if the domains across which they are being compared are semantically related. This is a fundamental question unaddressed by the model of [7]. We shall address this problem next.

2.3 Transferability Measurement

We now describe how to quantify transferability between domains in order to effectively exploit the cross-domain com-

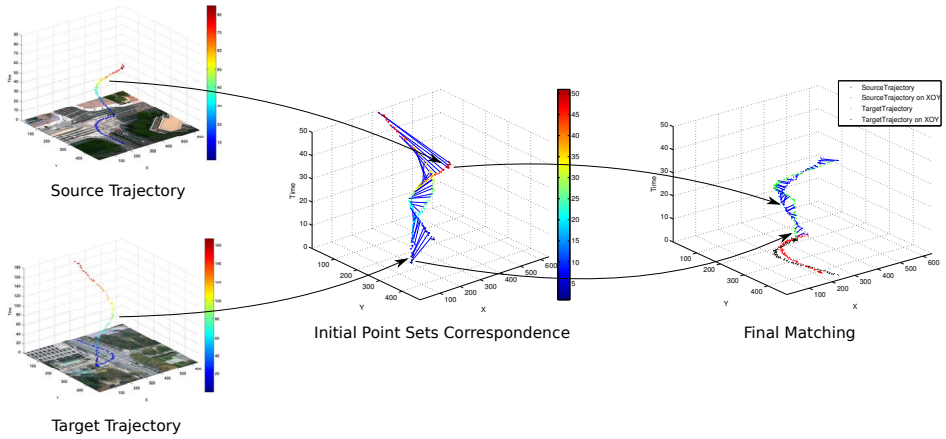


Figure 1: Cross-domain trajectory matching process. From left to right, two trajectory point sets are corresponded by the Hungarian algorithm, then a transformation is estimated based on the correspondence. The transformation is obtained by iterating this process.

parisons introduced in the previous section and hence achieve fully automatic sparse-shot cross-domain classification and anomaly detection. This question of "from where" to transfer is a known hard problem in transfer learning [16]. Relating one domain to an irrelevant domain typically results in worse performance than no transfer at all (negative transfer), and avoiding this is crucial. Insofar as the "from where" to transfer problem has been addressed [16], it typically requires labeled data in the target and source domain. Importantly, we will avoid making this assumption here, as relaxing this impractical requirement significantly increases the usefulness of such a system.

Assume there are $s = 1, \ldots, S$ available source domains. For each of these we have learned a collection of $c = 1, \ldots, C_s$ motion event models in the form of GMMs $\{g_{s,c}(\mathbf{x})\}_{c=1}^{C_s}$. Each source domain motion model may have a semantic label c if the goal is target domain classification. Now given a target domain t with a set of trajectories $T^t = \{g_t(\mathbf{x})\}$ observed online, we determine the most relevant source domain s^* for transfer to the current domain t by matching the distribution of trajectory-cluster distances within in the source ($\mathcal{H}_{ss}$) and across the target-source mapping ($\mathcal{H}_{ts}$):

$$\begin{aligned} \mathcal{H}_{ts} &= \mathcal{H}_{i \in T^t}(\min_{c_s} D^{\mathbf{H}^*}(g_{s,c_s}, g_i)), \\ \mathcal{H}_{ss} &= \mathcal{H}_{j \in T^s}(\min_{c_s} D(g_{s,c_s}, g_j)), \\ s^* &= \text{argmin}_s(\|\mathcal{H}_{ts} - \mathcal{H}_{ss}, \|). \end{aligned} \tag{5}$$

Here $\mathcal{H}(\cdot)$ indicates the histogram operator, $i \in T^t$ and $j \in T^s$ index target (g_i) and source (g_j) trajectories and respectively, g_{s,c_s} are the clustered motion patterns in the source domain, $\mathbf{H}^*$ is the optimal cross-domain transform (Eq. (3)), $\|\cdot\|$ is Euclidean distance, and the minimisation over c_s indicates matching clusters in source s. Thus domains are encoded by the *spread* of fits between trajectories and clusters, and matched by the similarity of those spreads.

This is derived from the intuition that two scenes which appear semantically similar to humans should have a similar distribution of motion. It is related to covariance descriptor methods in recognition [20] insofar as using the whole distribution of matches rather than quantifying a single goodness of fit. Two obvious alternatives are: (i) finding a rigid (rather than per-trajectory) similarity transform of all the target trajectories to sources, however this is computationally intractable and non-robust to e.g., piece-wise differences in scene layout; and (ii) finding the "best fit" source with minimum distance of individual target trajectories to the closest source patterns (Eq. (6))

$$s^* = \text{argmin}_s(\sum_{i \in T^t} \min_{c_s} D^{\mathbf{H}^*}(g_{s,c_s}, g_i)). \tag{6}$$

However this will *over fit* in that a complicated source scene with many different behaviours will always be the best fit for any target scene. In contrast, the proposed method is tractable and does not suffer from over fitting, as considering the full distribution of distances differentiates such domains.

As data is observed online in a target domain, we continually estimate and dynamically select the source domain for transfer via Eq. (5). Importantly, as we will show in the experiments, a good source domain can be selected with much less data than is required to build an effective local model in the target domain. Figure 2 illustrates the domain matching process.

2.4 Sparse-shot Anomaly Detection and Cross-Domain Event Classification

Given the domain-independent distance metric as explained in Section 2.2, and the optimal scene matching procedure as explained in Section 2.3, sparse-shot cross-domain event classification and anomaly detection is straightforward as follows. Trajectories represented as $g_t(\mathbf{x})$ in the target domain can be classified using the class c^* of their nearest source cluster:

$$c^* = \text{argmin}_c D^{\mathbf{H}^*}(g_t, g_{c,s^*}), \tag{7}$$

where s^* is the optimal source scene as determined by Eq. (5) using the data observed so far. Importantly, this allows classification in the target domain without requiring any annotations.

For anomaly detection, we consider the (similarity invariant) distance of $g_t(\mathbf{x})$ from the nearest cluster in the chosen

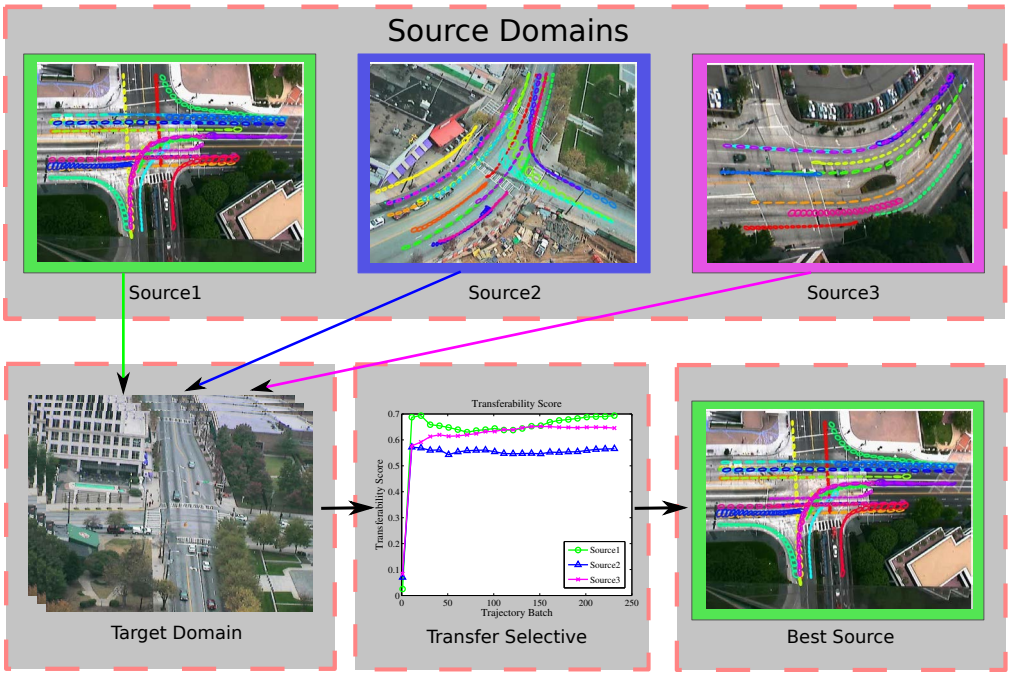


Figure 2: Source domain selection procedure: Compare sparse target domain trajectories to source domain clusters online. Select source domain by metric in Eq .(5).

source domain s^*:

$$D_t = \min_c D^{\mathbf{H}^*}(g_t, g_{c,s^*}). \quad (8)$$

Anomalous trajectories are flagged as those with distances D_t above a threshold θ_{th}. By quantifying abnormality relative to a selected semantically similar source domain s^*, significantly better anomaly detection can be obtained than by detecting anomalies against a local model built online using sparse/insufficient online observations. This is because the sparse target domain data can be used more effectively to match source domains online than to construct a good local model from scratch in the target domain.

3. EXPERIMENTS

Datasets: A motivating application of our framework is anomaly detection and event classification in surveillance videos captured by UAV platforms with far-field view. Therefore the NGSIM dataset [2] which is mainly taken by fixed cameras from a far field of view are good candidates in being representative of UAV videos. We evaluate our contributions on four scenes from NGSIM dataset: Lankershim 2 (LC2), Lankershim 4 (LC4), Peachtree 1 (PC1), and Peachtree 3 (PC3). These cover a variety of view angles and scene types, see Table 1 and Figure 3.

Preprocessing and Settings: For each scene, we extract all available trajectories (Table 1). We then over-cluster trajectories (Section 2.1; using $C_0 = 80$ as this is significantly more than the number of typical motion patterns) followed by self-tuning spectral clustering to merge motion models into representative clusters (Table 1). Then each trajectory cluster is represented by a corresponding motion pattern in the form of a GMM (Section 2.1) as illustrated in Figure 3. We test the performance of sparse-shot anomaly detection and classification on all 4 scenes (domains) in a leave one dataset out protocol. That is, we evaluate each dataset in turn as a target while considering the other there datasets as source domains.

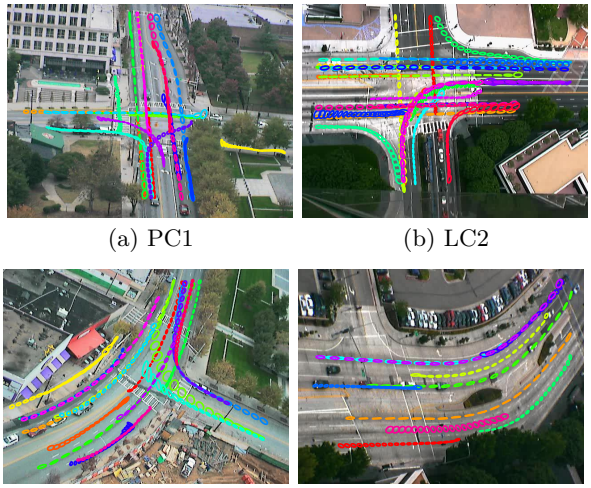
Alternative Models: We compare the following three models: (1) **Direct Transfer** by fixing each source domain in turn as the source; (2) **Local** by building a local model online with limited data (only for anomaly detection, not classification since annotation is assumed unavailable). This is the conventional approach to classification and anomaly detection [6, 13] generalised to online learning. For online learning we process target domain trajectories in chunks and build an updated motion model after observing every N additional trajectories, using this model to interpret the next chunk of trajectories; (3) **Baseline** by brute-force transfer, aggregating motion models from all available source domains. This provides a baseline for transfer anomaly detection and classification, but without source selection. Trajectories in the target domain are compared with motion models from all available source domains. (4) **Best Fit Transfer** is the simplest source domain selecting method. We select the source domain with minimal (transformed) distance of individual target trajectories to source clusters (Eq. (6)) after each batch of observed trajectories. (5) **Selective Transfer** is our full selective domain-transfer model. After each batch of input, we compute the transferability metric Eq. (5), and use the selected source to interpret observed trajectories.

3.1 Evaluation and Results

In these experiments, we evaluate the ability of our framework to select source domains and classify and detect abnormal motion events. Since there is no clear ground-truth for domain selection, we evaluate domain selection by way of whether the selected domain provides effective classifica-

Scenes	Frames	Rate	Resolution	View	Anomalies	Number of Trajectories	Learned Clusters
PC1	29918	10 f/s	640x480	$45-60^{\circ}$	1	2317	19
LC2	21700	10 f/s	640x480	Nadir	3	2412	28
PC3	29918	10 f/s	640x480	$45-60^{\circ}$	1	1468	19
LC4	20950	10 f/s	640x480	$45-60^{\circ}$	3	2444	10

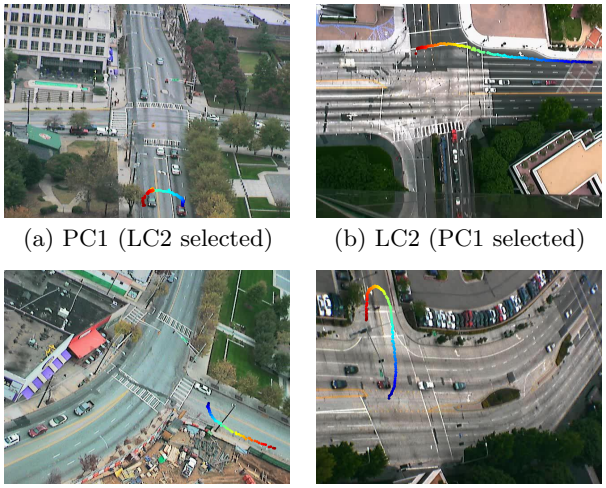
Table 1: Statistics and pre-processing results of each scene (domain).



(a) PC1 (b) LC2

(c) PC3 (d) LC4

Figure 3: Learned motion patterns (trajectory clusters) for each scene.



(a) PC1 (LC2 selected) (b) LC2 (PC1 selected)

(c) PC3 (LC4 selected) (d) LC4 (PC3 selected)

Figure 5: Illustration of abnormal events detected in each scene. Color indicates time.

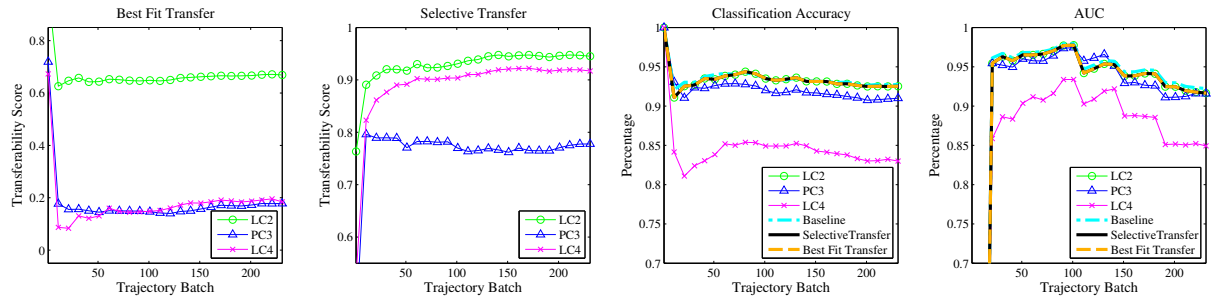
tion and detection. For abnormality detection, we manually annotated abnormal events in each scene (see Table 1 for statistics) which included events such as pedestrian jaywalking, u-turns and swerving. Good models should rank anomalies higher than typical motion events. To evaluate anomaly detection performance, we therefore compute the receiver operating characteristic (ROC) curve, which reflects true positive versus false positive rate as detection threshold θ_{th} is varied. This is then summarised by area under the curve (AUC) metric. For classification, we manually labeled events in each scene into three categories (turn-left, turn-right, go-straight). Classification performance is then evaluated by simple accuracy for each scene.

Source Selection for Classification: The results for direct and selective transfer methods are summarised in Figure 4. The first and second column show the source selection transferability metric for best fit and selective transfer respectively as a function of observed trajectory batch in the target dataset. In each case, both source-selection metrics converge to a consistent winner, very quickly relative to the length of the dataset (Figure 4, one line / source above the others quickly and consistently). However as expected, the best fit metric consistently prefers the most complex dataset (LC2), whereas only selective transfer metric selects a different source in each case, showing the selectivity of our metric.

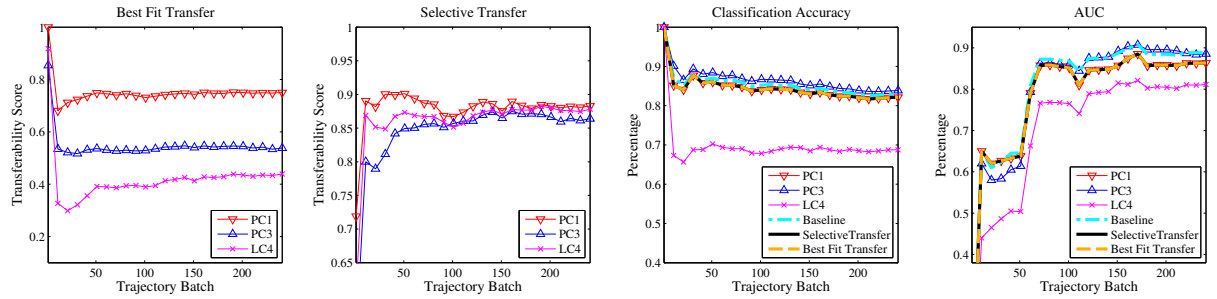
The third column shows the classification accuracy for each approach (three direct transfer conditions with coloured symbols, brute-force transfer (Baseline) in cyan dash-dot, best fit transfer in orange dash and our selective transfer framework in bold black). Considering the direct transfer conditions, each source dataset is sometimes worst by a significant margin. Meanwhile the brute-force baseline and best fit transfer mechanisms are also worst or near worst in some cases. In contrast, across the diverse combinations of sources and targets, our selective transfer framework is usually best (PC1) or near best (PC3, LC2 and LC4) overall. Importantly, our selective transfer metric consistently avoids the worst source (unlike best fit for LC4 and PC3), and is robust in case where the brute-force baseline is seriously poor (LC4). These results reflect both the serious risk of negative transfer and our framework's robustness to it.

Source Selection for Anomaly Detection: Anomaly detection performance is shown in Figure 4, fourth column. Again, in each case our selective transfer framework is best (PC3) or near-best (PC1, LC2, LC4) compared to fixed source domains transfer. Here the brute-force baseline performs closer to selective transfer, but an noticeable margin is still present in the PC3 case, where best fit also selects the worst source. Some detected abnormal events are illustrated in Figure 5, along with the selected source domain for each target. PC1 and LC2 are estimated to exhibit mutual transferability as well as PC3 and LC4. This is understandable given the straight nature of the first two scenes and the curvy nature of the latter two scenes. This source selection allows more effective anomaly detection. For example the U-turn and swerving driving in PC1 and LC2 respectively are ranked more highly as anomalies with LC2 and PC1 as the respective sources than they would be with PC3 and LC4 (which are intrinsically more curvy) as the sources.

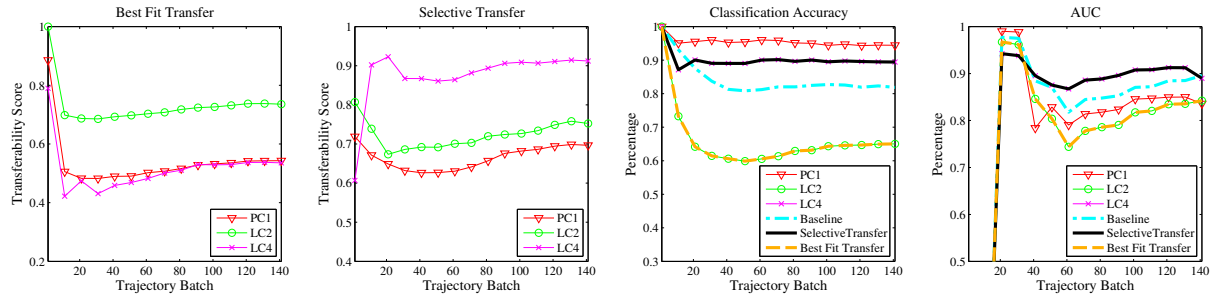
Sparse Data Stability of Source Selection: We next ask how stable is source selection in the rapid deployment



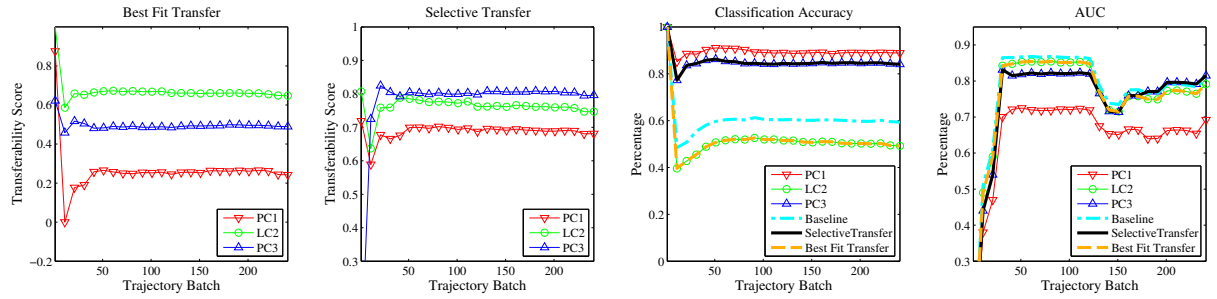
(a) PC1



(b) LC2



(c) PC3



(d) LC4

Figure 4: Cross-domain scene understanding. Rows: Target scenes. Columns: (1st) Source selection metric. (2nd) Classification accuracy. (3rd) Anomaly detection AUC.

/ very sparse target data context of interest, and how this compares to building a local model online. To test this we evaluate the detection of each target domain anomaly, embedded in a test set consisting of the 100 adjacent typical trajectories. We vary the size of a learning window (from 1 to 250 trajectories in batches of 10) ahead of test set in the data stream – effectively controlling how much data the local model has to learn typical behaviours, and how much data the transfer model has to select a suitable source scene. The results in Figure 6 show that our selective transfer framework (bold black) performs reasonably despite the extremely sparse data, selecting the best source in the 2 cases where the margin between best and worst is significant (PC3 and LC4). We note that in the two cases where selective does not make the best choice, it will eventually do so given enough data (PC1 and LC2 in Fig. 4). Compared to brute-force

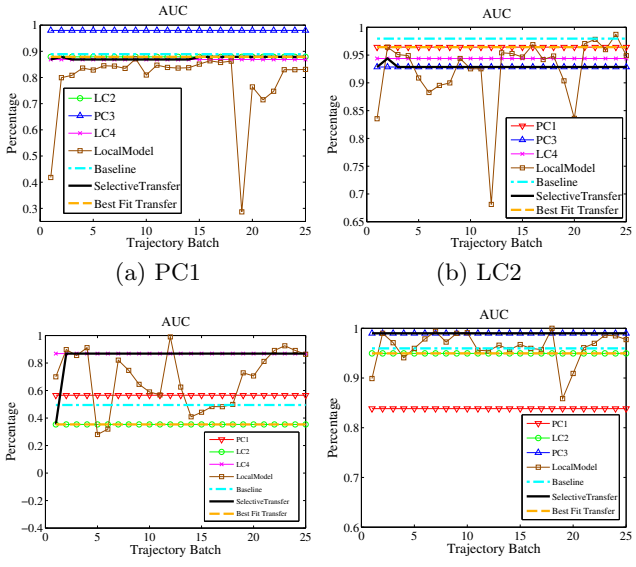


Figure 6: Anomaly detection with sparse data: Comparing constructing a local model with sparse data against using this data for domain selection.

and best fit transfer, selective transfer is in each case same or better in 3 datasets and worse in only one.

The most conventional strategy of building a local model online (brown) generally performs poorly, and importantly is very unstable. This is because with statistically insufficient training data, the rank of the anomaly varies dramatically as the particular samples included in the growing training set vary. Importantly, and in contrast to this instability, the source selection is quite stable even with such sparse inputs – performing consistently from as few as 10 observed trajectories. This highlights the important conclusion that sparse target domain data is much more effectively used for computing selective transfer to well understood domains than for building a weak statistically insufficient local model.

Discussion It is worth noting that a key aim of selective transfer is to avoid negative transfer by selecting source models which are suitable for interpreting target events. Previous methods such as [7] have considered transfer from one scene to another, but not how to deal with multiple sources of varying relevance. For this reason it is not possible to pose a direct comparison to [7], since it depends on how this would be generalised to make use of multiple sources. If it used one specific source, then it would roughly correspond to our single source conditions (aside from our technical improvements, mentioned in Section 2.2). If it aggregated all the source data together, it would roughly correspond to our brute-force baseline condition.

We have seen that best fit transfer falls down in non-selectively preferring the most complex scene. Meanwhile, brute-force transfer is intrinsically limited in the long term, as aggregating multiple sources increases over-fitting monotonically. Consider the case of anomaly detection: as many more source scenes are added to the pool, eventually some scene in which every behaviour is normal has been added. Now every target domain track – even abnormal ones – can be well explained by some source domain data, and anomaly detection is poor. Clearly this misses the point of context: abnormality is context dependent according to the semantics of the scene. Correctly determining the context in which an event should be interpreted is exactly what is achieved by our selective transfer mechanism.

4. CONCLUSION

We proposed a novel framework for cross-domain traffic scene understanding via motion model transfer. By learning models of a batch of source domains offline, we can do cross-domain sparse-shot anomaly detection and classification in a new scene. Crucially, we introduce a robust domain similarity criterion that enables robust domain-transfer by finding the most relevant source domain among a heterogeneous batch. Selecting a well learned source scene turns out to be a much more effective use of sparse local data than learning a local model. These results are an important contribution toward the topical goals of achieving re-locatable and hence scalable surveillance models. In future work we aim to further refine the computation of scene transferability, and will also develop this framework toward application to dynamic UAV surveillance.

5. REFERENCES

[1] D. H. Douglas and T. K. Peucker. Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. *Cartographica: The International Journal for Geographic Information and Geovisualization*, 10(2):112–122, Oct. 1973.

[2] Federal Highway Administration. Next generation simulation (ngsim) dataset. `http://ops.fhwa.dot.gov/trafficanalysistools/ngsim.htm`.

[3] A. Goshtasby. Image registration by local approximation methods. *Image Vision Comput.*, 6(4):255–261, 1988.

[4] J. Hershey and P. Olsen. Approximating the kullback leibler divergence between gaussian mixture models. In *ICASSP*, 2007.

[5] T. Hospedales, S. Gong, and T. Xiang. Video behaviour mining using a dynamic topic model. *International Journal of Computer Vision*, 98:303–323, 2012.

[6] W. Hu, X. Xiao, Z. Fu, D. Xie, T. Tan, and S. J. Maybank. A system for learning statistical motion patterns. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 28(9):1450–1464, 2006.

[7] S. Khokhar, I. Saleemi, and M. Shah. Similarity invariant classification of events by kl divergence minimization. In *ICCV*, 2011.

[8] J. F. Kooij, G. Englebienne, and D. M. Gavrila. A non-parametric hierarchical model to discover behavior dynamics from tracks. In *ECCV*, 2012.

[9] H. Kuhn. The Hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97, 1955.

[10] J. Li, S. Gong, and T. Xiang. Learning behavioural context. *International Journal of Computer Vision*, 97(3):276–304, 2012.

[11] J. Lin. Divergence measures based on the shannon entropy. *IEEE Transactions on Information Theory*, 37(1):145–151, 1991.

[12] R. Mehran, A. Oyama, and M. Shah. Abnormal crowd behavior detection using social force model. In *CVPR*, 2009.

[13] B. T. Morris and M. M. Trivedi. Learning and classification of trajectories in dynamic scenes: A general framework for live video analysis. In *AVSS*, 2008.

[14] B. T. Morris and M. M. Trivedi. A survey of vision-based trajectory learning and analysis for surveillance. *IEEE Transaction on Circuits and Systems for Video Technology*, 18(8):1114–1127, 2008.

[15] J. Nocedal and S. Wright. *Numerical optimization*. Springer-Verlag, 2nd edition, 2006.

[16] S. J. Pan and Q. Yang. A survey on transfer learning. *IEEE Transsaction on Knowledge and Data Engineering*, 22(10):1345–1359, Oct. 2010.

[17] J. Prokaj, X. Zhao, and G. G. Medioni. Tracking many vehicles in wide area aerial surveillance. In *CVPR Workshops*, 2012.

[18] I. Saleemi, L. Hartung, and M. Shah. Scene understanding by statistical modeling of motion patterns. In *CVPR*, 2010.

[19] A. Sodemann, M. Ross, and B. Borghetti. A review of anomaly detection in automated surveillance. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews,*, 42(6):1257–1272, 2012.

[20] O. Tuzel, F. Porikli, and P. Meer. Region covariance: A fast descriptor for detection and classification. In *ECCV*, 2006.

[21] X. Wang, X. Ma, and E. Grimson. Unsupervised activity perception by hierarchical bayesian models. In *CVPR*, 2007.

[22] L. Zelnik-Manor and P. Perona. Self-tuning spectral clustering. In *NIPS*, 2004.

[23] G. Zen, E. Ricci, and N. Sebe. Exploiting sparse representations for robust analysis of noisy complex video scenes. In *ECCV*, 2012.