Contents lists available at ScienceDirect

J. Vis. Commun. Image R.

journal homepage: www.elsevier.com/locate/jvci

Generalization of feature embeddings transferred from different video anomaly detection domains ☆

Fernando P. dos Santos *, Leonardo S.F. Ribeiro, Moacir A. Ponti

Institute of Mathematical and Computer Sciences (ICMC), University of São Paulo (USP), São Carlos, SP 13566-590, Brazil

ABSTRACT

Detecting anomalous activity in video surveillance often suffers from limited availability of training data. Transfer learning may close this gap, allowing to use existing annotated data from some source domain. However, analyzing the source feature space in terms of its potential for transfer of learning to another context is still to be investigated. This paper reports a study on video anomaly detection, focusing on the analysis of feature embeddings of pre-trained CNNs with the use of novel cross-domain generalization measures that allow to study how source features generalize for different target video domains. This generalization analysis represents not only a theoretical approach, can be useful in practice as a path to understand which datasets allow better transfer of knowledge. Our results confirm this, achieving better anomaly detectors for video frames and allowing analysis of transfer learning's positive and negative aspects.

1. Introduction

An anomaly detection algorithm infers a model that is able to discriminate between a normal pattern and abnormal ones. In this context, *learning* means inferring a function $f : X \rightarrow Y$ from a training set of examples $x_i \in X$, in which X is an input space from a given domain, and $Y = \{-1, +1\}$ is the output space, which in the case of anomaly detection refers to either "normal" or "anomaly". It is common to have knowledge mostly (or only) about the normal data and, therefore, most techniques models only what is considered normal in a given problem, and therefore considering any event that deviates from such normality as anomalous. Learning then means inferring a function using input data composed of observations from a single, normal, class [1]. In terms of video data, this approach is employed in surveillance of human crowds [2–4], pedestrian detection [5–7], and analysis of directions (human or vehicle motion) [8]. But even when considering complex systems, it is essential to have sufficient data and an adequate input space to be able to create an accurate detector.

In this context, real world problems become increasingly challenging when it comes to meeting the requirements of (annotated) data availability assumptions. First, the amount of available data

collected from some task is often sufficient only to the same problem or domain [9,10]. Second, many algorithms, in particular those with a large number of parameters to be learned such as deep learning methods, need a large amount of labeled data [11]. However, annotating large amounts of data for learning can be expensive [12,13]. With these considerations, there is a strong incentive to investigate techniques that can reduce the need for new labels and data [14]. *Transfer learning* can be a viable tool in this direction, such as demonstrated by [15] in the context of pedestrian detectors, and by [16] for anomaly detection in video. The goal of transfer learning is to allow solving new problems using knowledge previously acquired from other similar solutions [17]. In this learning context, the source dataset or domain supposedly provides sufficient knowledge to complement or even describe some target dataset or domain [18]. Therefore, the main challenge in transfer learning can be seen as to correlate the distribution of training data from a source to the distribution of test data from a target [19]. However, if the source and target are similar in their domains, data with similar distributions, it is expected that the same classifier performs similarly across them [20].

Transfer learning has been widely used in many different applications, such as image representation [21], facial attribute classification [22], network traffic [23], medical diagnosis [24], and pattern recognition [25]. In problems of video anomaly detection, either natural or urban scenarios, the assumption that similar problems have similar data distribution may not hold true due to factors such as variable illumination, camera perspective, and amount of clutter in the scene [5,26]. These factors contribute to

the distancing between the activities and visual content of different domains. More importantly, the definition of anomaly itself may differ. Although in general an anomaly is an observation containing an abnormal pattern [27], this can be linked to different events as, for example, the appearance of a clandestine boat for the application of water surveillance, or a car parking on a pedestrian boardwalk in a traffic surveillance scenario. Consequently, detecting anomalies in video requires an arduous task of studying which attributes are relevant to well model the activities, allowing systems to distinguish normal from unusual ones. Consider the task of anomaly detection in surveillance videos and any two distinct domains A and B: in this scenario, one can train a recognition system using a source domain A, in which $f : A \rightarrow Y$ may yield a reliable system for such domain. But it is not certain if the same function would be reliable when used within the context of some target domain B. Intuitively, all samples from B would anomalous, i.e. $f(b_i) = +1$ for all $b_i \in B$, leading the detector to fail for the target data. The challenge of transfer learning here is to design systems capable of performing *consistently*, even when considering cross-domain or cross-dataset scenarios, such as training on A and detecting on B. Fig. 1 illustrates this scenario in which two distinct domains training sets (A_{tr} and B_{tr}) are used to identify normal and anomalous activity of a single test domain (A_{te}). Our hypothesis is that, by using adequate transfer learning techniques, a new source domain (BA_{tr}) can be found, closing the gap between the data distributions of A and B.

In this paper we investigate the *generalization* of feature spaces extracted via a pre-trained Convolutional Neural Network (CNN) – which does not require additional labels – within the task of detecting anomalous activity in videos. CNNs have been successful in many computer vision tasks, being considered to be intelligent feature extraction modules that offer flexibility and a good level

of cross-domain transfer learning [17,28–30], even on video recognition tasks [3]. Investigating transfer learning in this scenario is relevant because CNN models are known to require large amounts of labeled training data in order to converge and to have any true learning guarantee; in practice many datasets do not have enough samples to allow training of deep networks from scratch. Leveraging pre-acquired knowledge is therefore a must and we find it important to highlight that recent research [31] has shown that theoretical learning guarantees are achieved when CNNs are trained with large datasets such as ImageNet [32].

We aim to understand the benefits of using CNN-based feature embeddings coupled with simple transfer learning methods in the task of anomaly detection. We designed experiments on transferring an anomaly detector's knowledge within: (i) the cross-domain feature embeddings; (ii) cross-domain Principal Component Analysis (PCA) [33]; and (iii) Transfer Component Analysis (TCA) [34]. More than designing a framework for transfer learning within the task of anomaly detection, we introduce a novel evaluation approach regarding generalization of feature embeddings between distinct domains and present the performance of our systems within these new metrics. Our contributions are then two-fold: (i) we developed a framework for transfer learning applied in the task of anomaly detection in videos; and (ii) we designed a novel evaluation approach regarding generalization of feature embeddings.

2. Method

In order to evaluate how well some feature space has the potential to generalize for a target dataset, we use an experimental setup that considers several different domains, each coming from different video anomaly detection datasets. Fig. 2 depicts this setup: all datasets are individually mapped to a feature space using the same pre-trained VGG-19 [35] network. For every domain pair (source A, target B), the training set of A is used to train an anomaly detector model (One Class Support Vector Machine (OC-SVM) [36]), which is then employed on the test set of B. Afterwards, we evaluate the generalization between A and B and the impact of transfer learning methods on this generalization.

Three scenarios were investigated: (i) without any data transformation; (ii) with PCA applied across domains; and (iii) with TCA. The next sections detail each step: the feature extraction (Section 2.1) and transfer learning methods (Section 2.2), as well as the generalization evaluation metric (Section 2.3).

2.1. Feature extraction

Feature extraction is a fundamental step that directly influences the result of recognition systems — a bad descriptor choice can significantly degrade performance and accuracy. With the great number of feature extraction methods available and the particularities of each database, finding the relevant descriptor can become a trial-and-error task [37]. Data-driven approaches to feature extraction are a recent trend, of which deep learning was particularly shown to produce suitable representations [38]. One of the main
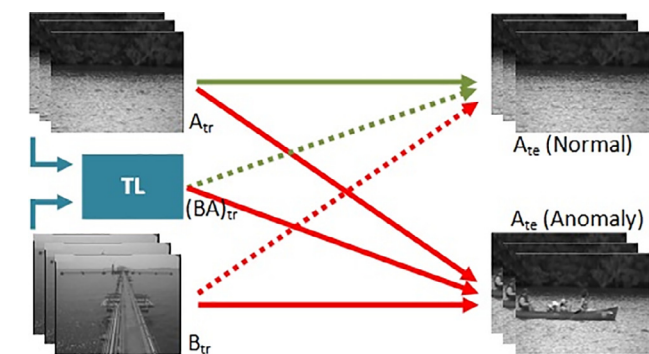

Fig. 1. Likely scenario in cross-domain application of video anomaly detection. The green arrows represent normal events detection and red arrows identify anomalous occurrences from two distinct scenarios (sources A and B); notice how normal events on the target domain are identified as anomalous due to the domain gap. It is expected that applying transfer learning methods (TL) between those two domains will significantly improve normal events detection on the target domain (represented by dashed flows). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)
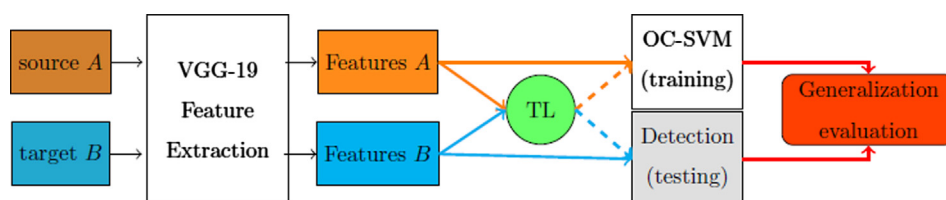

Fig. 2. Experimental setup: both source and target domains feature spaces embedding are independently computed via the same deep network model, then the source A is used to train a One Class SVM, while target B is tested on this trained model. Transfer learning (TL) can be used to transform such spaces before training/testing (indicated in dashed lines).

advantages presented by methods of feature learning in relation to handcrafted extraction is the generalization of the feature space produced for data not seen within the same visual domain [39,40]. Among current deep learning techniques, CNNs are widely used to compute feature space representations by sharing information and internal connections [41].

The VGG-19 CNN [35] is widely used as pre-trained models due to its simple architecture: a composition of convolutional layers with 3×3 sized filters. The motivation of this structure is that two sequential 3×3 filters have an effective receptive field of a 5×5 and, with the addition of more rectification layers, the decision function is more discriminative. This concept can be expanded to replace 7×7 filters with three filters 3×3. After a sequence of convolutional layers (generally composed of the 3×3 filters) and max-pooling, the top of VGG-19 is composed of fully connected layers which aim to provide probabilities for trained classes. Each layer provides a new descriptor that can be used to describe shapes and edges (lower layers) and texture and semantics (higher layers) [28], each one with a number of predefined attributes.

We use this architecture as a feature extractor, each frame of the selected domains is forward-passed through a network with its weights pre-trained using the ImageNet dataset [32] (and used as-is). To ensure compatibility with the pre-training phase, each frame has its resolution reduced to 224×224 and the last network layer is discarded due to its tie with the ImageNet classification task; the feature set is then a result of the second-to-last layer (FC-4096), a 4096-dimensional feature vector.

2.2. Transfer learning

For our baseline, the feature space obtained in the second-to-last layer in VGG-19 is applied in two circumstances: without pre-processing and with dimensionality reduction. Considering two domains (A and B) and their respective training set (A_{tr} and B_{tr}) and test set (A_{te} and B_{te}), the cross-feature occurs with direct inversion of test sets in relation to their source domain. Therefore, the anomaly detection performed in A_{te} uses the information contained in B_{tr} and A_{tr} is the basis for B_{te}. With this setup and without pre-processing, all 4096 attributes available by the VGG-19 layer are considered in the evaluation of the **first scenario**.

For the **second scenario**, we investigate how the classic space projection technique PCA [33] may contribute to transfer learning. We apply PCA on the original feature space (A_{tr}) and learn a projection matrix Θ of this space to a subset of its eigenvectors. We experiment then using the same projection Θ on the feature space of another domain (B_{te}) and evaluate how well does learning trained over A perform on B given both spaces were projected over A's eigenvectors.

Finally, in the **third scenario** TCA is applied for the task of anomaly detection over distinct domains. TCA, introduced by Pan et al. [34], is motivated by the assumption that common factors exist between different domains. The goal of performing TCA is to project both feature spaces into a new, common space where the distance between samples from distinct domains is small and data variance is kept large (this latter objective being the same as designed by classic PCA). The solution showed by Pan et al. [34] is to formulate TCA as a variant of Kernelized-PCA, where data is centralized over both domains, which the Gram Matrix is defined as the composition of gram matrices, where the subscripts indicate the source (S) or target (T) domains and $K_{S,S} = X_S X_S^t$:

$$K = \begin{bmatrix} K_{S,S} & K_{S,T} \\ K_{T,S} & K_{T,T} \end{bmatrix} \tag{1}$$

To build this matrix, TCA needs at least some training data from the second domain and K is weighted to highlight intra-domain and diminish inter-domain dissimilarities, stimulating the eigen

decomposition to better capture existing variance across-domains instead of in each individually. Using this approach, TCA is designed to find eigenvectors and eigenvalues in a combined feature space, therefore acting upon the assumption that common factors do exist between domains and finding a linear transformation into a space that highlights those common factors.

2.3. A Generalization metric for cross-domain feature spaces

The field of machine learning has long dealt with the idea of developing theoretical guarantees and support for what is called "learning" within the context of each algorithm; the most stable theory comes in the form of Statistical Learning Theory (STL) [42,43]. STL has since its introduction been widely used to assess the quality of studies within machine learning field and, more specifically, to support the mathematical proofs that guarantee Support Vector Machines (SVM) generate optimum classifiers [44].

As one of the major contributions of this study, we hereafter propose a new metric to evaluate cross-domain transfer learning systems using SLT tools. Inspired on the evaluation of supervised learning models, which proved invaluable to researchers working to design such systems, we aim then to contribute to the field of cross-domain transfer learning by asking: how can one measure generalization of a feature space produced by some method? One of the main concepts that drive STL is generalization. In supervised learning, generalization is a divergence that measures how well a classifier performance with unseen data is *consistent* with its performance on training (seen) data. It can be mathematically expressed as:

$$|R_{emp}(f_n) - R(f_n)|, \tag{2}$$

where $R(f_n)$ is the true risk (expectancy of loss) of a classifier f_n over "all data", also called expected risk, and $R_{emp}(f_n)$ is the risk of the same f_n, but evaluated over the training set (the empirical sample), called empirical risk. The idea of true risk is purposefully abstract (being an intractable quantity), but it nonetheless serves its goal of highlighting the importance of not losing ourselves amid metrics of accuracy and cost over training data, metrics that may not paint the bigger picture of how well a system works [44].

Our view is that the evaluation of transfer learning systems applied to anomaly detection cannot rely solely on performance metrics such as Receiver Operating Characteristic (ROC), and derived metrics such as Area Under the ROC Curve (AUC) and Equal Error Rate (EER); if we aim to measure how well a system trained within one domain's feature space performs on a dissimilar domain's space, the idea of generalization presents itself as a great fit. We propose then two ways to measure it: (i) Partial Cross-domain Feature Space Generalization (Partial CDFG, or G_{part}); and (ii) Complete Cross-domain Feature Space Generalization (Complete CDFG, or G_{comp}), defined as:

$$G_{part}\left(f_n^A\right) = \left| \underset{x \in X^A}{R(f_n^A)} - \underset{x \in X^B}{R(f_n^A)} \right| \tag{3}$$

$$G_{comp}\left(f_n^A, f_n^B\right) = \frac{1}{2}\left(\left| \underset{x \in X^A}{R(f_n^A)} - \underset{x \in X^B}{R(f_n^A)} \right| + \left| \underset{x \in X^B}{R(f_n^B)} - \underset{x \in X^A}{R(f_n^B)} \right| \right) \tag{4}$$

where f_n^A and f_n^B are classifiers found by a classification algorithm as part of Empirical Risk Minimization applied over, respectively, domain A and domain B; the expression $\underset{x \in X^A}{R(f_n^A)}$ denotes the risk of classifier f_n^A over the feature space X^A and $\underset{x \in X^B}{R(f_n^A)}$ denotes the risk of the same classifier f_n^A over the feature space of the second

domain, X^B; the mirrored definition being valid for $\begin{smallmatrix}R(f_n^B)\\x\in X^B\end{smallmatrix}$ and $\begin{smallmatrix}R(f_n^B)\\x\in X^A\end{smallmatrix}$. Even though the real risk is intractable, it could be approximated by validation or test sets (data that is specially separated to represent unseen data) from each domain. Finally, transferring those ideas to the context of anomaly detection, we could define the empirical risk in this formulation as the evaluation of either *AUC* or *EER*, both classic metrics for anomaly detection tasks.

Given two domains (*A* and *B*) and their respective feature spaces (X^A and X^B), G_{part} and G_{comp} are meaningful metrics if:

- The bias (set of admissible functions) of the classification/detection algorithms are the same (e.g. same parameters on a SVM training setup);
- The spaces X^A and X^B are composed of the same set of descriptors for both domains;
- The transfer learning or domain mapping method should have no prior knowledge of test data on either domain.

Evaluating with CDFG affords us a multi-leveled analysis of feature spaces and transfer learning systems. G_{part} is a good representation of how well training over a domain *A* is well-suited or adapted to work over samples from domain *B* with consistent performance. It is however "one-way" regarding domain, i.e we are looking at the mapping or transferring characteristic only in the $A \rightarrow B$ direction. It is, regardless of this limitation, relevant to the analysis, especially when used in conjunction with G_{comp}. While G_{part} is partial to the chosen descriptors and how each domain is particularly well represented by such descriptors, G_{comp} is a better measure of the quality of the transfer system itself and its robustness when tested over different contexts and pairs of domains.

We are going to hereafter introduce three particular analysis levels afforded by our novel metrics. To compare results by pairs of methods (classification algorithm and transfer learning techniques, indicating by the subscripts α and β), we selected classifiers obtained through the principle of empirical risk minimization for each methodology, denoted as f_α and f_β. One could use inequalities, such as expressed bellow:

$$G_{part}\left(f_\alpha^A\right) < G_{part}\left(f_\beta^A\right) \tag{5}$$

With this inequality satisfied, one could claim that method α is capable of generalizing well from domain *A* to domain *B*. One can also verify the G_{part} metric from the "opposite direction" and assess if the α methodology is also better than β at generalizing from *B* to *A*, as expressed by the inequality:

$$G_{part}\left(f_\alpha^B\right) < G_{part}\left(f_\beta^B\right) \tag{6}$$

Hence, evaluating the G_{part} on both directions gives us an understanding of the first level of generalization: how well the space obtained from one domain is applicable to another and how this applicability is captured by the chosen methodology. These comparisons do not assess directly the transfer method, being influenced and capturing well aspects regarding the representation capability of each feature space. To obtain a more precise and rigorous analysis of the transfer learning method itself, we should compare using the G_{comp} metric:

$$G_{comp}\left(f_\alpha^A, f_\alpha^B\right) < G_{comp}\left(f_\beta^A, f_\beta^B\right) \tag{7}$$

However, the best use of our metrics come from applying each G_{part} and G_{comp} at the same time, given that the G_{comp} can be influ-

enced by high discrepancy between the two G_{part} that compose it. It is primal then that all three comparisons are taken into account in the assessment of any two competing methods.

3. Experiments

We recall the experimental setup depicted in Fig. 2. First, we evaluate the feature spaces with anomaly detection. With these results, we apply our generalization metrics to compare the performance of domains and methods used. In the following sections we describe the datasets and detail the experiments.

3.1. Datasets

For transfer learning and feature generalization experiments, we use seven anomaly detection videos/datasets (natural and urban scenarios) differing in several aspects, including frame resolution, amount of training frames, illumination conditions, perspectives, and presence of clutter.

Natural scenarios (water surveillance activities):

- **Canoe**: a video with 1050 frames of 240×320 pixels, which 200 are separated for training. It has scenery of nature with a river in the center and a canoe which invades the waters indicating the anomaly [45];
- **Boat-River:** similar to Canoe, but with higher resolution (576×740 pixels) and different perspective. It has only 80 training frames and the majority of frames with anomalies in a single video [46];
- **Boat-Sea:** similar to Canoe and Boat-River, but with occlusion in the scenes, making it difficult to correlate with the first two videos. Boat-Sea is composed of a single video of 576×720 resolution, in which the first 100 frames are for training and the remaining ones for tests [46].

Urban scenarios (traffic and pedestrian activities):

- **UCSD Pedestrian 1 (Ped1):** it consists of videos of footpath, in which the presence of pedestrians are considered normal, and abnormal activity includes cyclists, skaters, and others. UCSD Ped1 contains 34 videos for training and 36 videos for tests with 158×238 pixels of resolution [47];
- **UCSD Pedestrian 2 (Ped2):** similar to Ped1, UCSD Ped2 contains 16 videos for training and 12 videos for tests with 240×360 pixels of resolution. However, the placement of the cameras is different between the datasets [47];
- **Belleview:** it has frames with 240×320 pixels in a single video, with images of a road intersection in which anomalies are characterized by vehicle conversions and the standard behavior by the straight-line pass [46];
- **Train:** it is also a single video (the first 800 frames of 386×288 are used in training) where the illumination varies rapidly due to the passage of the train through the tunnels. The anomalies are composed by the movement of the people inside the wagons [46].

All frames from Canoe, Boat-River, Boat-Sea, and Train were converted to grayscale via: *0.299R + 0.587G + 0.114B*. The other urban datasets are originally in grayscale. Examples from all videos/datasets are presented in Fig. 3. Since the urban datasets are more complex and need to be described via motion/direction attributes, we expect the feature embeddings and transfer learning methods to work better within the Natural scenarios.
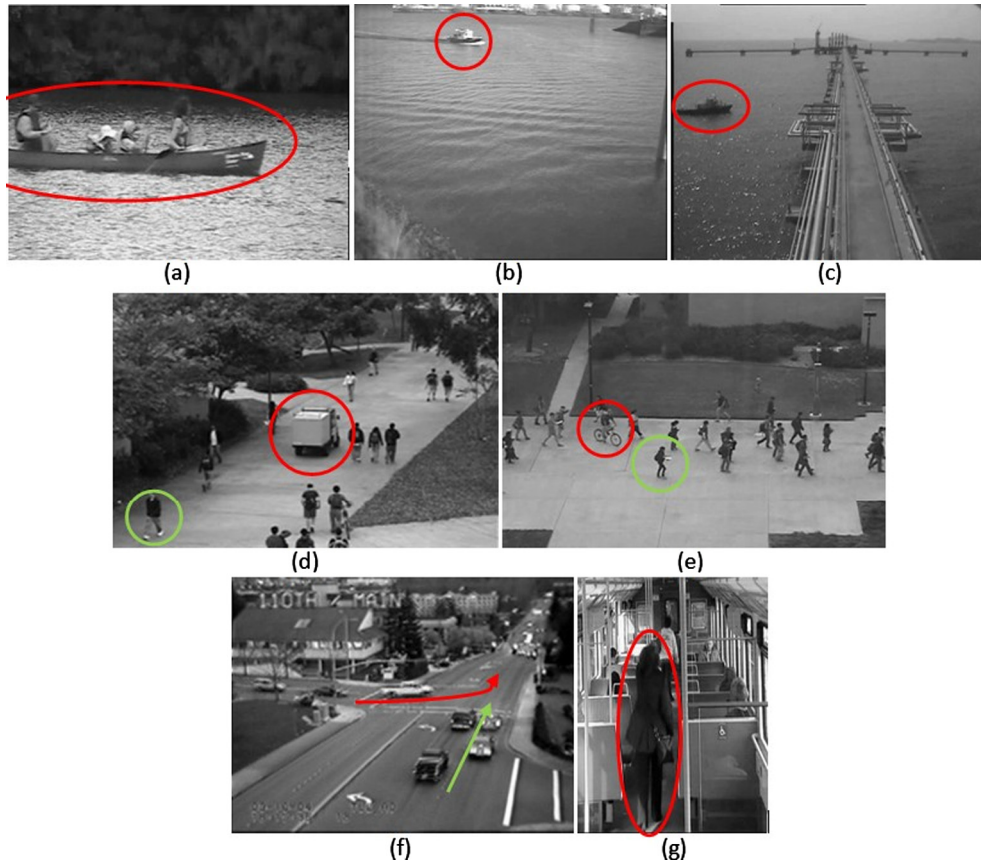
Fig. 3. Samples of test frames from: (a) Canoe; (b) Boat-River; (c) Boat-Sea; (d) UCSD- Ped1; (e) UCSD-Ped2; (f) Belleview; and (g) Train. Anomalous events are represented in red (boats, trucks, cyclists, vehicle conversions, and passenger movement). Examples of normal events are in green (pedestrians and straight-line pass). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

3.2. Anomaly detection, parameters and evaluation

The Area Under the Curve (AUC) and Equal Error Rate (EER) are used to evaluate the anomaly detection in frame level. That means the model must predict which frames contain at least one anomaly and compare them with their respective ground-truth to determine the false positive rate (FPR) and true positive rate (TPR). The EER is the value on the ROC Curve in which $FPR = 1 - TPR$. As an anomaly detector, we used OC-SVM [36] that estimates support vectors having only positive labels in training set by including regularization to single out outliers. When the linear kernel is used, a hyper-sphere is drawn to cover the training samples, allowing an amount of outliers v in $(0, 1]$. In our approach, we used $v = 0.25$.

For our generalization measure to be validated, we performed three experiments: first using the original feature embedding (Full VGG-19) with 4096 features; second, by applying PCA to these spaces selecting the 80 principal components (this value was chosen since Boat-River video has the smallest training set, 80 training examples, limiting the PCA analysis); third, the same dimensionality (80) is used in TCA method using the RBF kernel. In the original VGG-19, only the source training set is used to infer the model, which is then used to identify abnormal events using the target test set. For PCA, the transformation computed for the source training set is used to transform and select features for the target test set. In order to apply TCA, the matrix merged with the two training sets (source and target) is used to compute the transformation matrix, which is then used to select the principal components in target test set. With the results of the metrics (AUC and EER) the generalization measure is then computed. In that sense, we evaluate from a lower to a higher level of knowledge transfer. Table 1

shows the PCA variance obtained with the videos/datasets: values near 1.0 in all sets indicate that there was no loss of relevant data in the created space with 80 features, due to the similarity among.

4. Results and discussion

Table 2 presents the anomaly detection results of three experiments for natural scenarios: using the original VGG-19 feature embedding (Full VGG-19), after transformation with PCA (reduction to 80 dimensions), and applying unsupervised transfer learning by TCA (also with 80 dimensions). Considering the AUC and EER as performance metrics in the comparison of TCA with the other two approaches, it is observed that the TCA is better in 6 pairs (66.6% of total) tested, mainly when the source domain is Boat-River. In the natural scenarios, the average AUC across all TCA sets was 86.68%, meanwhile for Full VGG-19 was 70.47%. As expected, the PCA space performance is out-performed by TCA, both in the number of pairs (1 vs. 6), in the average AUC (69.52%

Table 1
PCA variance with 80 principal components.

Dataset	Variance
Canoe	0.9996
Boat-River	1.0
Boat-Sea	0.9999
UCSD-Ped1	0.9992
UCSD-Ped2	0.9998
Belleview	0.9997
Train	0.9998

Table 2
Anomaly detection in natural scenarios (%).

Source → Target	Full VGG-19		PCA		TCA	
	AUC	EER	AUC	EER	AUC	EER
Canoe → Canoe	**92.63**	**12.65**	63.97	39.66	71.66	32.49
Boat-River → Canoe	92.75	12.65	53.61	48.1	**99.1**	**3.04**
Boat-Sea → Canoe	92.75	12.65	85.44	18.56	**97.5**	**4.64**
Boat-River → Boat-River	63.24	36.75	74.35	25.64	**90.59**	**9.4**
Canoe → Boat-River	**63.24**	**36.75**	50.42	49.57	61.11	38.88
Boat-Sea → Boat-River	64.52	35.47	61.96	38.03	**94.87**	**5.12**
Boat-Sea → Boat-Sea	54.97	46.15	**97.01**	**9.89**	91.37	16.48
Canoe → Boat-Sea	55.0	46.15	83.39	26.37	**86.99**	**19.79**
Boat-River → Boat-Sea	55.2	46.15	55.54	43.96	**86.95**	**21.91**
Average	70.47	28.37	69.52	33.3	**86.88**	**16.86**

vs. 86.88%), and average EER (33.3% vs. 16.86%). Additionally, there are expressive results of TCA in relation to Full VGG-19 and PCA: Boat-Sea → Boat-River has an improvement (30.35% to Full VGG-19) and Boat-River → Boat-Sea in 31.41% in relation to PCA. Therefore, as expected, the space provided by TCA overcomes Full VGG-19 and PCA in natural scenarios.

Also, considering the AUC and EER as performance metrics in urban scenarios (see Table 3), it is highlighted that Full VGG-19 is better in 7 pairs (16 pairs in total), mainly when the target domain is Ped2 or Belleview. However, the difference between the averages of Full VGG-19 and TCA is practically negligible: 63.84% vs. 62.8% in AUC and 39.41% vs. 40.51% in EER. Unlike the results with natural environments, urban domains with PCA presented positive results. It is important to emphasize that the concept of anomalies between these domains is very different, implying that the transfer learning should not be totally transferred (negative transfer). Hence, considering only domains with the same meaning of anomalies (Ped1 and Ped2), TCA stands out in relation to Full VGG-19 and PCA in averages of AUC and EER.

Although the transfer learning by TCA be superior to original space (Full VGG-19) and PCA, those metrics (AUC and EER) are not enough to guarantee the feature space generalization. Analyzing the results in isolation gives an imprecision due to the great variety of performances achieved. For these reasons, our generalization metrics offers a more detailed and reliable comparison if one methodology overcomes other. Evidently, generalization does not depend only on the techniques, but also on the similarity

among domains. In Fig. 4 is presented a frame of the Boat-River video, pointed out as anomalous, which erroneously was detected as "normal" using Full VGG-19 (Boat-Sea → Boat-River). However, applying TCA the anomaly was detected (Boat-Sea → Boat-River).

4.1. Transferred features generalization

Based on the metrics used to anomaly detection (AUC and EER) on generalization measure proposed in this paper, we evaluated the performance of the feature spaces provided by Full VGG-19 cross-domain, PCA cross-domain, and transfer learning by TCA. First, we evaluate the generalization at the first level, G_{part}. The results using the Inequations (5) and (6) are presented in Table 4. In general, the transfer learning method TCA is superior in the two metrics evaluated in the G_{part}. Considering only AUC, the average of all cases (12 sets) of TCA was 8.47%, with Full VGG-19 in 22.43%, and PCA in 17.27%. The same occurs with EER, being the best rate for TCA with 8.1%. In terms of similarity between domains, Canoe and Boat-Sea are very close in the feature space mapping, in both directions TCA performs with high transfer rates. There is also great applicability of the feature spaces in contexts of different anomalies, Belleview and Ped1, in which the transfer learning is more significant from Ped1 to Belleview. It is also observed that Ped1 offers high learning rates for Ped2; however the inverse does not occur in the same intensity. Another highlight is the higher performance of the PCA when compared to Full VGG-19, demonstrating that the dimensionality reduction increases the

Table 3
Anomaly detection in urban scenarios (%).

Source → Target	Full VGG-19		PCA		TCA	
	AUC	EER	AUC	EER	AUC	EER
Ped1 → Ped1	50.91	51.4	**71.46**	**35.17**	62.94	39.68
Ped2 → Ped1	50.82	51.46	**64.01**	**39.13**	60.39	41.7
Belleview → Ped1	51.77	50.75	**76.12**	**30.56**	58.86	45.89
Train → Ped1	53.42	51.75	60.65	39.72	**71.02**	**33.66**
Ped2 → Ped2	**80.34**	**26.26**	55.24	44.13	74.16	33.26
Ped1 → Ped2	**80.18**	**26.25**	56.95	46.14	67.06	38.54
Belleview → Ped2	**80.88**	**26.26**	69.46	34.63	65.16	38.01
Train → Ped2	**81.81**	**25.69**	61.77	41.89	50.11	52.51
Belleview → Belleview	68.91	33.47	50.54	51.38	**72.63**	**32.24**
Ped1 → Belleview	**68.67**	**33.45**	56.22	45.45	68.39	35.25
Ped2 → Belleview	**68.73**	**33.47**	60.42	40.63	65.24	39.12
Train → Belleview	**69.1**	**32.92**	54.36	49.31	68.65	34.35
Train → Train	53.97	47.2	**57.67**	**42.84**	51.88	51.47
Ped1 → Train	54.02	46.73	**57.75**	**46.16**	53.98	43.96
Ped2 → Train	54.13	46.67	55.47	49.0	**55.56**	**42.68**
Belleview → Train	53.85	46.84	50.63	51.46	**58.86**	**45.89**
Average	**63.84**	**39.41**	59.92	42.97	62.8	40.51

Fig. 4. Example of same frame from Boat-River considered normal with Full VGG-19 (left) and anomalous with TCA (right).

Table 4
Partial cross-domain feature space generalization (G_{part}) (%).

Source → Target	Full VGG-19		PCA		TCA	
	AUC	EER	AUC	EER	AU	EER
Boat-River → Canoe	29.51	24.1	20.74	22.46	**8.51**	**6.36**
Boat-Sea → Canoe	37.78	33.5	11.57	8.67	**6.13**	**11.84**
Canoe → Boat-River	29.39	24.1	13.55	9.91	**10.55**	**6.39**
Boat-Sea → Boat-River	9.55	10.68	35.05	28.14	**3.5**	**11.36**
Canoe → Boat-Sea	37.63	33.5	19.42	13.29	**15.33**	**12.7**
Boat-River → Boat-Sea	8.04	9.4	18.81	18.32	**3.64**	**12.51**
Ped2 → Ped1	29.52	25.2	**8.77**	**5.0**	13.77	8.44
Belleview → Ped1	17.14	17.28	25.58	20.82	**14.27**	**10.47**
Ped1 → Ped2	29.27	25.15	14.51	10.97	**4.12**	**1.14**
Belleview → Ped2	11.97	7.21	18.92	16.75	**7.47**	**5.77**
Ped1 → Belleview	17.76	17.95	15.24	10.28	**5.45**	**4.43**
Ped2 → Belleview	11.61	7.21	**5.18**	**3.5**	8.92	5.86
Average	22.43	19.6	17.27	14.0	**8.47**	**8.10**

performance during cross-feature. These data contradict the isolated analysis from Tables 2 and 3, implicit the importance of our generalization metrics.

The generalization in the first level excludes more complex and pertinent aspects to the generated spaces. The G_{part} analysis is not enough to verify the two domains in a unique model, except in cases where there will be only contribution from one domain to other, without the need of the inversion from source and target. This scenario is noticeable in situations where the source is composed of large amounts of data and, therefore, it is sufficient to provide information to itself, not requiring auxiliary domains or prior learning. In more complex and accurate scenarios, G_{comp} offers a deeper analysis of the proposed transfer learning model. In this approach, results in Table 5, TCA offers even more generalization in relation to other two methods. Among similar domains, the latent space created by TCA is highly applicable: Boat-River and

Boat-Sea with 3.57%; Boat-River and Canoe with 9.53%; and Ped1 and Ped2 with 8.95%. Even in domains with different anomalies, the performance gain is evidenced (Ped2 and Belleview with 8.19%).

For a full feature generalization, the three inequalities (5, 6, and 7) must be satisfied. This level guarantees that the G_{comp} is contemplated without one G_{part} compensating the other. In our experiments, it is observed that there is compensation in two approved cases of G_{comp} with TCA: (Ped1, Ped2) and (Belleview, Ped2). Although Ped1 and Ped2 have the same concept of anomalies, the position of the cameras hinders the direct transfer learning, requiring preprocessing methods to facilitate the use of previously acquired knowledge. Despite the urban scenario, the concept of anomalies between Belleview and Ped2 is different both semantically and visually: Belleview targets vehicles conversion, while Ped2 anomalies are related to the presence of vehicles on the scene.

Table 5
Complete cross-domain feature space generalization (G_{comp}) (%).

Datasets	Full VGG-19		PCA		TCA	
	AUC	EER	AUC	EER	AUC	EER
(Canoe, Boat-River)	29.45	24.1	17.14	16.18	**9.53**	**6.37**
(Canoe, Boat-Sea)	37.70	33.5	15.49	10.98	**10.73**	**12.27**
(Boat-River, Boat-Sea)	8.79	10.04	26.93	23.23	**3.57**	**11.93**
(Ped1, Ped2)	29.4	25.2	11.6	7.99	**8.95**	**4.79**
(Ped1, Belleview)	17.5	17.6	20.4	15.6	**9.86**	**7.45**
(Ped2, Belleview)	11.79	7.21	12.05	10.12	**8.19**	**5.82**

4.2. Negative transfer

A major concern in transfer learning methods is to apply only the acquired knowledge that favors the improvement of the task for the new target domain. For this purpose, it is important to evaluate if the source domain is sufficiently related to target domain so that the transfer does not fail, causing the negative transfer [48]. The negative transfer is evidenced when the transfer learning method achieved a lower performance to a method that does not perform transfer learning [49]. In this context, G_{part} and G_{comp} should be applied to measure if the source domain or the methodology is suitable for a designated task.

Tables 6 and 7 present correlation results between videos/datasets of urban scenarios, more specifically between Train and the others (Ped1, Ped2, and Belleview). Train presents concept of

anomalies very different from the others, in which the dissimilarity between them (background and objects) are highly perceivable. By the analysis of G_{part}, it is observed that there is applicability of transfer learning only from Train → Ped2. Differently, the other methods (Full VGG-19 and PCA) performed better than TCA, characterizing a negative transfer scenario. Consequently, G_{comp} indicates that Train is not a suitable domain for Ped1, Ped2, or Belleview.

4.3. A closer look on the practical application of the generalization metric

The datasets studied in this paper have two levels of difficulty: the water surveillance videos are easier, depending mainly on the appearance of the objects on the frame, while the urban ones

Table 6
Partial cross-domain feature space generalization (G_{part}) (%): negative transfer.

Source → Target	Full VGG-19		PCA		TCA	
	AUC	EER	AUC	EER	AUC	EER
Train → Ped1	**0.55**	**4.55**	2.98	3.12	19.14	17.81
Train → Ped2	27.84	21.51	4.1	0.95	**1.77**	**1.04**
Train → Belleview	15.13	14.28	**3.31**	**6.47**	16.77	17.12
Ped1 → Train	**3.11**	**4.67**	13.71	10.99	8.96	4.28
Ped2 → Train	26.21	20.41	**0.23**	**4.87**	18.6	9.42
Belleview → Train	15.06	13.37	**0.09**	**0.08**	13.77	13.65

Table 7
Complete cross-domain feature space generalization (G_{comp}) (%): negative transfer.

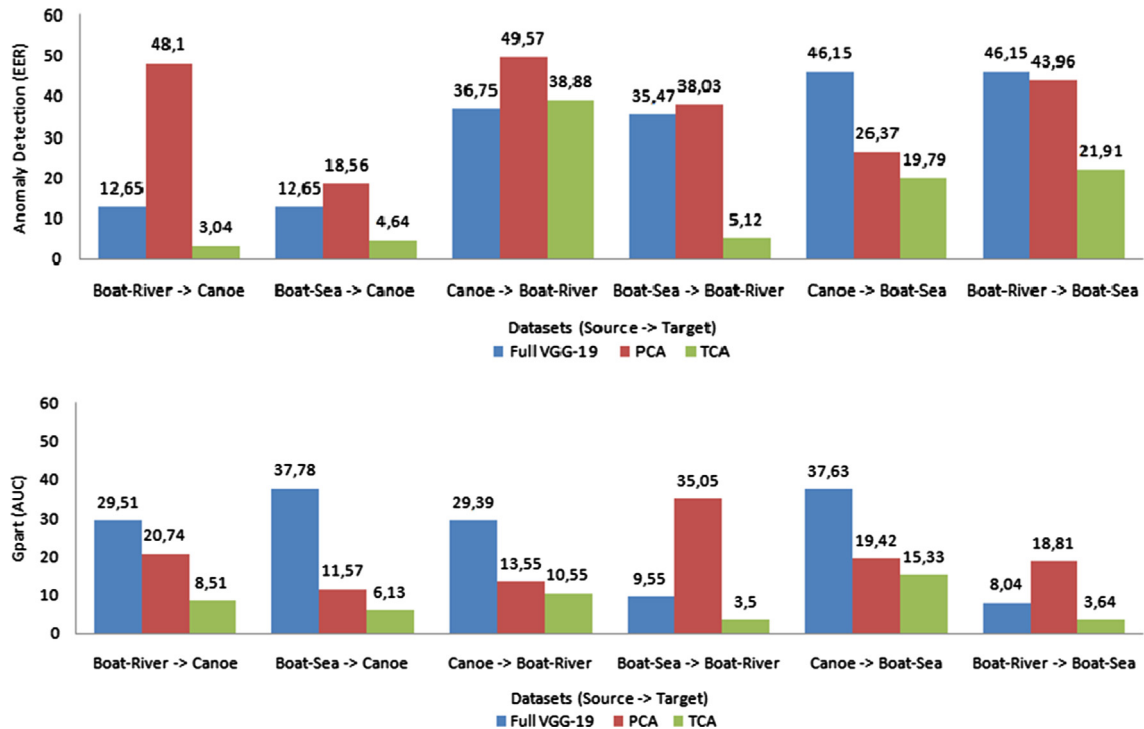| Datasets	Full VGG-19		PCA		TCA	
	AUC	EER	AUC	EER	AUC	EER
(Train, Ped1)	**1.83**	**4.61**	8.35	7.06	14.1	11.0
(Train, Ped2)	27.0	21.0	**2.17**	**2.91**	10.2	5.23
(Train, Belleview)	15.1	13.8	**1.7**	**3.28**	15.3	15.4

Fig. 5. Generalization from different water surveillance domains: (top) anomaly detection in EER; and (bottom) G_{part} in AUC.
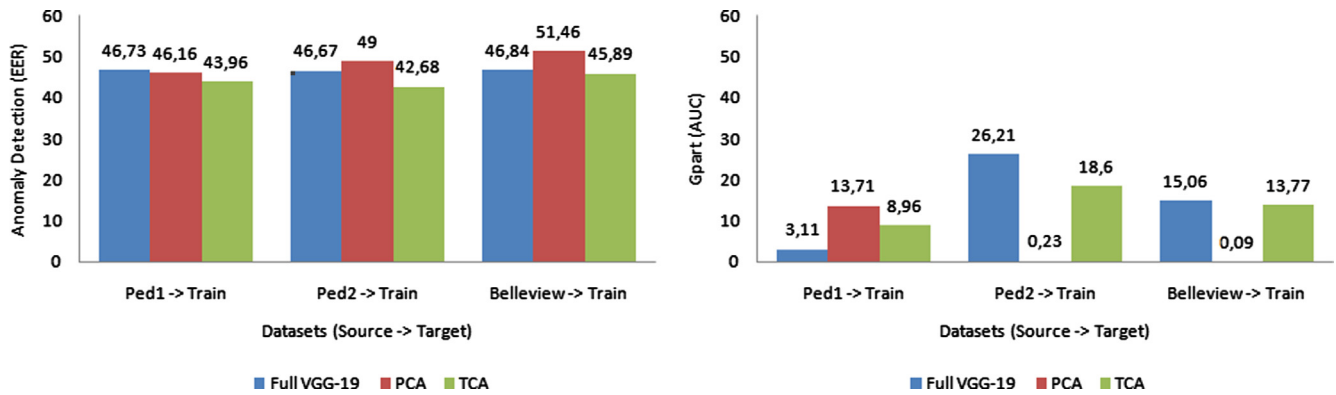
Fig. 6. Generalization from different source datasets to Train. The left barplots shows the EER (%) from anomaly detection results, while the right barplots from the G_{part} values.

depend on more complex attributes such as motion and orientation (not explicitly captured via the employed VGG-19 features), and have a larger training set. In any case, the use of the full feature embedding shows similar results for a fixed target domain B, and any source domain A. As an example, for the Boat-River dataset, training an anomaly detector on any source domain produces an EER of ∼36%. However, the generalization measure is not the same and indicates the best potential for transfer learning via TCA.

In Fig. 5, it is possible to see that, when evaluating the original feature embedding using both anomaly detection and the G_{part}, the dataset with the best (lower) generalization is a better potential as a source domain. If so, then the TCA method shows the best results. In the case of Boat-River (Fig. 5- top), the lowest EER are for Boat-Sea, which indeed, present the best result after application of TCA. Similar results are observed for Boat-Sea, for which the lowest G_{part} is Boat-River, which also produces good improvement after applying TCA. For Train video as target, in Fig. 6, the Ped1, Ped2, and Belleview were not possible to improve the results using transfer learning in the original feature embedding, which is clear when TCA often showed worse G_{part} when compared to either PCA or the VGG features.

4.4. Running time analysis

In our experiments, the frame per second processing rate was also analyzed. We consider two groups: water surveillance videos; and only UCSD datasets. The first one examines the average among all combinations of Canoe, Boat-River, and Boat-Sea. The second group considered the average when using datasets UCSD Ped1 and Ped2 only. Those experiments were grouped to evaluate scenarios considering videos with different duration. After running all experiments, we discarded the highest and the lowest recorded times to avoid outliers. In Table 8 we present the processing rates of frames per second (FPS), as well as the range of frames processed in the experiments for each group of videos. All experiments were performed on the same machine, configured with an Intel(R) Core (TM) i7-7700 K CPU, 64 GB of RAM and a GeForce GTX 1080 Ti GPU. It is important to highlight that feature extraction using feed-forward through neural networks was performed using the GPU

hardware while PCA, TCA, and SVM computations were performed on CPU.

Note that, overall, PCA presents the highest FPS, since it only has to process the training samples from the source domain and, because it compacts the space, it allows for a faster OC-SVM detection. When more video frames are used for Transfer Learning (see UCSD row of Table 8), the computational running time performance of TCA is severely degraded due to the need of keeping a model with frames from both source and target videos. Again, PCA allows a fast projection into a lower dimensionality. In any case, even a rate of 4 FPS is not unfeasible considering our system was not optimized for detection, and that running TCA on GPU or using multiple cores of the CPU would significantly improve this performance.

5. Conclusion

This paper proposes a cross-domain generalization metric that allows evaluation of feature embeddings, indicating the potential for transfer learning. As our results indicate, when using CNN-based features, the TCA performance stands out and it is often accompanied by better generalization levels. This is a simple approach that allows a guideline for the use of off-the-shelf feature extraction tools, boosting the performance of anomaly detection methods even when there is no additional data from the target domain that we are interested in solving.

We present experimental evidence that such generalization measures are useful in practice as a manner of understanding which datasets can be used as source or additional knowledge, and the ones to avoid, in order to describe video frames improving discrimination between normal and anomalous activity. This is important because it allows to use unsupervised or semi-supervised methods.

Transfer learning from video activity and other computer vision tasks is still a matter of future investigation. The proposed measures can be explored in the context of choosing which feature extraction method better suits some task, or to merge different datasets in order to accumulate a larger training set, and therefore increase learning guarantees.

Conflict of Interest

The authors declared that there is no conflict of interest.

Acknowledgment

Table 8
Frame per seconds (FPS) using the two investigated approaches considering the stages: feature extraction (CNN), space transformation (PCA or TCA) and detection (OC-SVM).

Video (# of analyzed frames)	Full VGG-19	PCA	TCA
Water (395 to 1162)	1256 FPS	2247 FPS	1754 FPS
UCSD (7110 to 20800)	47 FPS	1700 FPS	4 FPS

References

[1] V. Chandola, A. Banerjee, V. Kumar, Anomaly detection: A survey, ACM Comput. Surv. (CSUR) 41 (3) (2009) 15.

[2] H. Guo, X. Wu, S. Cai, N. Li, J. Cheng, Y.-L. Chen, Quaternion discrete cosine transformation signature analysis in crowd scenes for abnormal event detection, Neurocomputing 204 (2016) 106–115.

[3] Y. Hu, H. Chang, F. Nian, Y. Wang, T. Li, Dense crowd counting from still images with convolutional neural networks, J. Vis. Commun. Image Represent. 38 (2016) 530–539.

[4] R. Chaker, Z. Al Aghbari, I.N. Junejo, Social network model for crowd anomaly detection and localization, Pattern Recogn. 61 (2017) 266–281.

[5] M.J. Roshtkhari, M.D. Levine, An on-line, real-time learning method for detecting anomalies in videos using spatio-temporal compositions, Comput. Vis. Image Underst. 117 (10) (2013) 1436–1452.

[6] M.J. Hasan, J. Choi, A.K. Neumann, L.S. Roy-Chowdhury, Davis Learning temporal regularity in video sequences, in: Computer Vision and Pattern Recognition (CVPR), 2016 IEEE Conference on, IEEE, 2016, pp. 733–742.

[7] M. Ponti, T.S. Nazare, J. Kittler, Optical-flow features empirical mode decomposition for motion anomaly detection, in: Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on, IEEE, 2017, pp. 1403–1407.

[8] E. Epaillard, N. Bouguila, Proportional data modeling with hidden markov models based on generalized dirichlet and beta-liouville mixtures applied to anomaly detection in public areas, Pattern Recogn. 55 (2016) 125–136.

[9] Y. Aytar, A. Zisserman, Tabula rasa: Model transfer for object category detection, in: Computer Vision (ICCV), 2011 IEEE International Conference on, IEEE, 2011, pp. 2252–2259.

[10] T. Tommasi, F. Orabona, B. Caputo, Safety in numbers: Learning categories from few examples with multi model knowledge transfer, in: Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on, IEEE, 2010, pp. 3081–3088.

[11] H. Ravishankar, P. Sudhakar, R. Venkataramani, S. Thiruvenkadam, P. Annangi, N. Babu, V. Vaidya, Understanding the mechanisms of deep transfer learning for medical images, in: Deep Learning and Data Labeling for Medical Applications, Springer, 2016, pp. 188–196.

[12] L. Duan, D. Xu, I.W.-H. Tsang, J. Luo, Visual event recognition in videos by learning from web data, IEEE Trans. Pattern Anal. Mach. Intell. 34 (9) (2012) 1667–1680.

[13] X. Li, M. Fang, J.-J. Zhang, Projected transfer sparse coding for cross domain image representation, J. Vis. Commun. Image Represent. 33 (2015) 265–272.

[14] M. Long, Y. Cao, J. Wang, M.I. Jordan, Learning transferable features with deep adaptation networks, arXiv preprint arXiv:1502.02791.

[15] M. Wang, W. Li, X. Wang, Transferring a generic pedestrian detector towards specific scenes, in: 2012 IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2012, pp. 3274–3281.

[16] D. Xu, W. Ouyang, E. Ricci, X. Wang, N. Sebe, Learning cross-modal deep representations for robust pedestrian detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 5363–5371.

[17] J. Lu, V. Behbood, P. Hao, H. Zuo, S. Xue, G. Zhang, Transfer learning using computational intelligence: a survey, Knowl.-Based Syst. 80 (2015) 14–23.

[18] J. Yosinski, J. Clune, Y. Bengio, H. Lipson, How transferable are features in deep neural networks?, in: Advances in Neural Information Processing Systems, 2014, pp 3320–3328.

[19] J. Hu, J. Lu, Y.-P. Tan, Deep transfer metric learning, in: Computer Vision and Pattern Recognition (CVPR), 2015 IEEE Conference on, IEEE, 2015, pp. 325–333.

[20] E. Tzeng, J. Hoffman, T. Darrell, K. Saenko, Simultaneous deep transfer across domains and tasks, in: Computer Vision (ICCV), 2015 IEEE International Conference on, IEEE, 2015, pp. 4068–4076.

[21] M. Oquab, L. Bottou, I. Laptev, J. Sivic, Learning and transferring mid-level image representations using convolutional neural networks, in: Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on, IEEE, 2014, pp. 1717–1724.

[22] N. Zhuang, Y. Yan, S. Chen, H. Wang, C. Shen, Multi-label learning based deep transfer neural network for facial attribute classification, Pattern Recogn. 80 (2018) 225–240.

[23] G. Sun, L. Liang, T. Chen, F. Xiao, F. Lang, Network traffic classification based on transfer learning, Comput. Electr. Eng. (2018).

[24] B. Cheng, M. Liu, D. Zhang, D. Shen, A.D.N. Initiative, et al., Robust multi-label transfer feature learning for early diagnosis of alzheimers disease, Brain Imaging Behav. (2018) 1–16.

[25] J. Wang, H. Zheng, Y. Huang, X. Ding, Vehicle type recognition in surveillance images from labeled web-nature data using deep transfer learning, IEEE Trans. Intell. Transp. Syst. (2017).

[26] T. Hao, D. Wu, Q. Wang, J.-S. Sun, Multi-view representation learning for multi-view action recognition, J. Vis. Commun. Image Represent. 48 (2017) 453–460.

[27] F. Jiang, Y. Wu, A.K. Katsaggelos, Detecting contextual anomalies of crowd motion in surveillance video, in: Image Processing (ICIP), 2009 16th IEEE International Conference on, IEEE, 2009, pp. 1117–1120.

[28] A.S. Razavian, H. Azizpour, J. Sullivan, S. Carlsson, CNN features off-the-shelf: an astounding baseline for recognition, in: Computer Vision and Pattern Recognition Workshops (CVPRW), 2014 IEEE Conference on, IEEE, 2014, pp. 512–519.

[29] M. Sun, Y. Wang, T. Li, J. Lv, J. Wu, Vehicle counting in crowded scenes with multi-channel and multi-task convolutional neural networks, J. Vis. Commun. Image Represent. 49 (2017) 412–419.

[30] G.B. Cavallari, L.S.F. Ribeiro, M.A. Ponti, Unsupervised representation learning using convolutional and stacked auto-encoders: a domain and cross-domain feature space analysis, 31st SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI 2018), 2018.

[31] R.F. de Mello, M.D. Ferreira, M.A. Ponti, Providing theoretical learning guarantees to deep learning networks, arXiv preprint arXiv:1711.10292.

[32] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, et al., Imagenet large scale visual recognition challenge, Int. J. Comput. Vision 115 (3) (2015) 211–252.

[33] I.T. Jolliffe, Principal component analysis and factor analysis, in: Principal Component Analysis, Springer, 1986, pp. 115–128.

[34] S.J. Pan, I.W. Tsang, J.T. Kwok, Q. Yang, Domain adaptation via transfer component analysis, IEEE Trans. Neural Networks 22 (2) (2011) 199–210.

[35] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, arXiv preprint arXiv:1409.1556.

[36] Y. Chen, X.S. Zhou, T.S. Huang, One-class svm for learning in image retrieval, Image Processing, 2001. Proceedings. 2001 International Conference on, vol. 1, IEEE, 2001, pp. 34–37.

[37] Y. Xian, X. Rong, X. Yang, Y. Tian, Evaluation of low-level features for real-world surveillance event detection, IEEE Trans. Circ. Syst. Video Technol. 27 (3) (2017) 624–634.

[38] M.A. Ponti, L.S.F. Ribeiro, T.S. Nazare, T. Bui, J. Collomosse, Everything you wanted to know about deep learning for computer vision but were afraid to ask, in: 2017 30th SIBGRAPI Conference on Graphics, Patterns and Images Tutorials (SIBGRAPI-T), IEEE, 2017, pp. 17–41.

[39] S. Sarraf, G. Tofighi, Deep learning-based pipeline to recognize Alzheimer's disease using FMRI data, in: Future Technologies Conference (FTC), IEEE, 2016, pp. 816–820.

[40] T. Zhou, I. Icke, B. Dogdas, S. Parimal, S. Sampath, J. Forbes, A. Bagchi, C.-L. Chin, A. Chen, Automatic segmentation of left ventricle in cardiac cine MRI images based on deep learning, Medical Imaging 2017: Image Processing, vol. 10133, International Society for Optics and Photonics, 2017, p. 101331W.

[41] M. Liang, X. Hu, Recurrent convolutional neural network for object recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 3367–3375.

[42] V.N. Vapnik, An overview of statistical learning theory, IEEE Trans. Neural Networks 10 (5) (1999) 988–999.

[43] U. von Luxburg, B. Schölkopf, Statistical learning theory: Models, concepts, and results, Handbook of the History of Logic, Vol. 10: Inductive Logic, vol. 10, Elsevier North Holland, Amsterdam, Netherlands, 2011, pp. 651–706.

[44] R.F. Mello, M.A. Ponti, Machine Learning: A Practical Approach on the Statistical Learning Theory, Springer, 2018.

[45] P.-M. Jodoin, J. Konrad, V. Saligrama, Modeling background activity for behavior subtraction, in: Distributed Smart Cameras, 2008. ICDSC 2008. Second ACM/IEEE International Conference on, IEEE, 2008, pp. 1–10.

[46] A. Zaharescu, R. Wildes, Anomalous behaviour detection using spatiotemporal oriented energies, subset inclusion histogram comparison and event-driven processing, in: European Conference on Computer Vision, Springer, 2010, pp. 563–576.

[47] V. Mahadevan, W. Li, V. Bhalodia, N. Vasconcelos, Anomaly detection in crowded scenes, in: Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on, IEEE, 2010, pp. 1975–1981.

[48] L. Torrey, J. Shavlik, Transfer learning, in: Handbook of Research on Machine Learning Applications and Trends: Algorithms, Methods, and Techniques, IGI Global, 2010, pp. 242–264.

[49] S.J. Pan, Q. Yang, et al., A survey on transfer learning, IEEE Trans. Knowl. Data Eng. 22 (10) (2010) 1345–1359.