University of Modena and Reggio Emilia

XXXIV cycle of the International Doctorate School in
Information and Communication Technologies

Doctor of Philosophy dissertation in
Computer Engineering and Science

Prior Knowledge Exploitation and Transfer in Deep Learning Architectures

Angelo Porrello

Supervisor: Prof. Simone Calderara
PhD Course Coordinator: Prof. Sonia Bergamaschi

Modena, 2021

Review committee composed of:

Concetto Spampinato, University of Catania
Massimo Piccardi, University of Technology Sydney

To those I care about.

Contents

Chapter 1

Introduction

In the last decade, Deep Learning emerged as a hot topic and a disruptive tool in the fields of Machine Learning and Computer Vision. It builds upon a learning paradigm in which data (*e.g.*, videos acquired by surveillance cameras placed on a public road) play a crucial role. By leveraging a great number of data-points, it is possible to fit complex and human-like tasks (*e.g.*, recognizing abnormal actions in a video-stream) with impressive results. However, if data availability represents the source of the greatest strength of Deep Learning techniques, it also reveals the greatest weakness: the development of applications and services is indeed often restrained by such a requirement, as the acquisition and maintenance of a huge amount of data are expensive activities that require expert staff and equipment.

However, the design of modern Deep Learning architectures offers several degrees of freedom that can be exploited to mitigate the lack of training data, either partial or complete. The underlying idea is to compensate for it by incorporating prior beliefs that humans (specifically, those who control and guide the learning process) hold about the domain at hand. Indeed, intrinsic rules and properties extend far beyond training data and can often be identified and imposed on the learner. If we take image classification into account, the success of Convolutional Neural Networks (CNNs) over past solutions (such as Multi-Layered Neural Networks) can be mainly ascribed to such a practice. Indeed, the design principle of its fundamental building block (*i.e.*, the convolution between two 2D-signals) naturally reflect what we knew about images: in this regard, the correlations that subsist between neighborhood regions of the image provided a powerful insight for the development of effective models as CNNs still prove to be.

1.1 Research Statement

The ultimate aim of this thesis is the investigation and proposal of novel ways of modeling and injecting prior knowledge in Deep Learning applications. Importantly, we conduct such a discussion across the board: it focuses on several data domains (*e.g.*, images, videos, graph-structured data, etc.) and concerns different levels of the overall training pipeline. In particular, each chapter discusses a distinct application field or domain, in which the adoption of techniques leveraging prior knowledge has proven to be beneficial. Along with a comprehensive description of both settings and tools involved, this thesis presents extensive experimental results and ablation studies demonstrating the value of the techniques proposed in this research.

1.1.1 Organization

To guide the reader across this research, we now explain the organization of the rest of this thesis: the strategies that have been investigated can be broadly divided into three categories, reported in the following paragraphs.

Parameter-Based approaches A common way to introduce prior knowledge consists in limiting the space of feasible solutions to those regions reflecting geometrical properties of the data. This can be achieved by designing tailored neural layers, engineered to reflect some noticeable properties of the underlying domain. In this regard, this thesis analyzes two fields in which such a practice proves to be beneficial: namely, **graph classification** and **novelty detection**.

- Chapter 2 – mainly based on [142] – showcases the use case of graph classification: here, the examples come as complex and heterogeneous structures, composed by multiple nodes (vertices) coupled with additional prior information describing the interactions (edges) between each of them. Specifically, the main subject of this chapter is an approach that takes explicitly into consideration those interactions through a specific and novel layer placed inside the network.
- Chapter 3 deals with novelty detection – which aims at recognizing the occurrence of anomalous and novel events – and reports the discussion and findings of [1]: here, an entire auxiliary network is dedicated to modeling the regular traits of normal events. Remarkably, the original work introduces a prior knowledge on the latent space (*i.e.* a causal structure lying between

its latent variables): this is carried out through specific auto-regressive layers making up the above-mentioned auxiliary network.

Data-Driven approaches Alternatively, one can use standard neural building blocks and, differently, model prior knowledge in terms of the expertise of a neural network, which has been already trained and hence knows the underlying task. Here, the idea is to pin its output as a form of prior information we would like to maintain and then transfer its capabilities to another network, typically called the student network. Several works refer to this schema as the teacher-student paradigm and place it within the field of **Knowledge Distillation** [63]. We, therefore, discuss two potential applications of such a paradigm:

- Chapter 4 focuses on continual learning, a research field identifying all those approaches that mitigate the occurrence of *catastrophic forgetting* [130] while learning a sequence of tasks. Here, the prior knowledge we seek to transfer regards the beliefs of the network about examples of old tasks: in more details, Dark Experience Replay – the approach discussed in [23] and reported in Chapter 4 – uses old model's output responses to promote consistency with its past.
- Re-Identification is the main subject of Chapter 5: here, the task is to match images depicting the same identity (*e.g.*, a car or a pedestrian) captured from disjoint camera views. In particular, the chapter builds upon the intuition investigated by [143]: in short, the prior knowledge aimed to be transferred lies on the set of visual details concealed in many images of the given target. In this regard, we impose that leveraging only some of the available views is enough to recover the entire visual original content. We experimentally show that such an objective leads to more robust feature extractors.

Initialization-oriented approaches We finally assess a strategy that injects prior knowledge by providing a smart initialization of the network's parameters. Such a practice – often referred as *pre-training* – usually builds upon the introduction of a preliminary pretext task, which usually has the following characteristics: *i)* its design encodes some advantageous, general and reusable skills that can be easily transferred to the downstream task; *ii)* a huge volume of data can be leveraged, thus lowering the risk of overfitting. To have a clear picture, Chapter 6 presents a practical scenario where the approach outlined above provides performance gains: in particular, the chapter recall [197] and deals with the classification of satellite images.

Chapter 2

Graph Classification

Convolutional Neural Networks (CNNs) have been successfully applied in different domains, such as speech recognition [62], image classification [89], and video analysis [180]. In these domains, data can be described as a signal defined on a regular grid, whose underlying dimension can be 1d, 2d or 3d. One of the key aspects of CNNs is that such a regular structure makes it possible to exploit local and stationary properties of data. Moreover, the convolution operator, its behaviour being equivariant to translations, allows filters with a limited support on the input grid, leading to a significantly smaller number of parameters with respect to Fully Connected Networks.

However, we are surrounded by data lying on an underlying structure, which typically has an irregular and non-euclidean nature. This is the case, for instance, of document databases, 3D skeletal data, information from social networks and chemical compounds. In all these domains, the relationships among entities are more complex than in the case of a simple grid-like connectivity. Instead, graphs constitute better representation forms, because they model directly the topological structures of such data domains, through edge weights. For this reason, many efforts have recently been made [22, 84, 36] in an attempt to generalise CNNs for graph-structured data.

In this chapter we focus on signal classification in homogeneus graphs. In such context, each sample obeys a single $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ weighted graph, which reflects the prior knowledge we dispose about the physics and the structure of the underlying domain. Specifically, the point in which a sample differs from the others is represented by the value of each vertex in the graph. As in the case

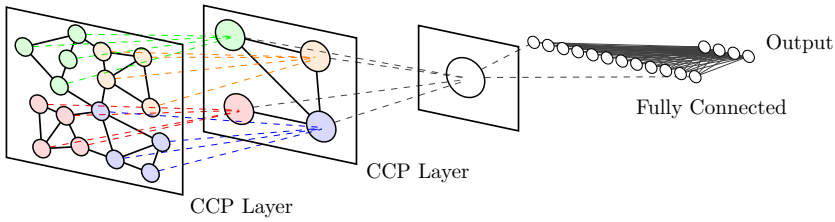


Figure 2.1: An overview of the proposed architecture. Multiple applications of the CCP layer lead to a multi-scale clustering of the input graph, exploiting both local and global properties during the information's flow from input to output. Finally, a Multi-Layer Perceptron classifies a global representation of the input signal, captured by a feature vector on a singleton graph.

of [67], we refer to each sample as a realisation of a signal on $\mathcal{G}$. The aim is to learn a function which maps each sample into the label space. By doing so, similarly to what CNNs do for images, at each step we shall exploit information coming from the neighbouring nodes.

In the following we introduce a tailored architecture, built by stacking multiple Convolutional Cluster Pooling (CCP) layers as depicted in Fig. 2.1. This layer firstly performs a clustering operation on the input graph, resulting in a coarser output graph, whose affinity matrix reflects relationships among clusters regressed at training time. By doing so, a good basis for building local receptive fields is achieved. Secondly, according to the neighbours' vision dictated by the first step, the layer selects for each cluster a fixed number of candidate nodes for the aggregation phase, and sorts them depending on a centrality-based rank within the cluster. In this respect, it is worth noting that weight sharing across the graph's neighbourhoods can be successfully exploited.

2.1 Preliminaries

Because of its generality and potential applications, the possibility to extend neural networks to deal with graph-structured data has become an active research area. Two main branches arise from the literature: **spectral methods**, which encode the graph structure using the Fourier Transform, and **spatial methods**, modelling the filtering operation through the creation of locally connected neighbourhoods.

Spectral approaches In general terms, spectral approaches take advantage of the fact that eigenvectors of the graph Laplacian span a space in which the convolution operator is diagonal [67]. Bruna*et al.* [22] exploited this property and defined a frequency filtering operation for neural networks. However, with such kind of formulation, it is not possible to relate the filtering operation within the spectral domain with the one performed in the vertex domain. In order to define localized linear transformations (*i.e.* operations also interpretable in the vertex domain [67]), Defferrard*et al.* [36] proposed the use of polynomial spectral filters, with a theoretical guarantee of k-localisation in space. In addition, they provided a recursive approximation of such filtering through Chebyshev polynomials, which prevent expensive computations needed by the Laplacian eigenvectors.

Spatial approaches The other branch concerns spatial methods, which directly model convolutions as a linear combination of vertices in a local neighbourhood. In this respect, the authors of Diffusion-Convolutional Neural Networks (DCNNs) [5] presented an approach in which feature vectors are spread according to the hop distance in a depth search tree, the latter having as parent root the node for which the operation has to be done. Kipf & Welling [84] proposed a fast and simple layer-wise propagation rule, which involves the use of normalized adjacency matrix. An interesting aspect of this method is how, from a spectral perspective, it may also be seen as an approximation of a localized first-order filter. Notably, the framework described by Monti*et al.* [132] led to a unified vision for all spatial approaches, in which the differences among different types of methods lie on the notion of the local coordinate system.

Relation with existing works Our model is to be considered a spatial approach, because we derive a convolution-like operation directly from the clustering step, the latter creating groups of spatially close vertices itself. Inspired by Deep Locally Connected Networks [22], we then assimilate the pooling operation with the filtering stage, providing a strategy to enable weight sharing across graph's clusters. Moreover, we propose a learnable multilevel strategy for graph coarsening, which may be performed directly during the learning process. On the latter point, our proposal differs from [36], where the Graclus multilevel clustering algorithm [158] has been used, the latter being performed during a pre-processing step. On this note, we were inspired by the work of Such*et al.* [176], who introduced graph embed pooling, a way to produce pooled graphs with a parametrizable number of vertices. However, our method is quite different in the computation of the

pooled vertices' feature maps. Indeed, while they consider output vertices as a weighted combination of all input vertices (where weights are given by clusters' memberships), we only sample a fixed number of vertices, and combine them according to learnable kernel's weights. Our spatial formulation builds on the concept that the weight sharing property can be inducted in a graph-oriented architecture, provided that a nodes-ordering criteria has previously been defined. A similar idea arised in PATCHY-SAN [136], in which a ranking procedure and a graph normalisation technique have been used to generate local receptive fields, resulting in an adjacency matrix for each selected node. This way, the authors managed to exploit structural and local properties of input graph very well. However, the authors did not address how intermediate sub-graphs should be merged and, consequently, how that procedure should be stacked on multiple layers. The latter point could make it difficult to capture global structures with the same effectiveness. Differently, our method generates receptive fields for entire clusters, enabling graph coarsening and a hierarchical architecture.

2.1.1 Hierarchical Graph Clustering

A graph $\mathcal{G}$ can be defined as an ordered pair $(\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is a set of $\mathcal{N}$ nodes and $\mathcal{E} \in \mathcal{V} \times \mathcal{V}$ a set of edges. Here, we are interested in classifying signals defined on an undirected and weighted graph, in which $\mathcal{E}$ can be described by a real symmetric matrix $\mathcal{A} \in \mathbb{R}^{\mathcal{N} \times \mathcal{N}}$ which, for each couple of vertices $\mathcal{V}_i$ and $\mathcal{V}_j \in \mathcal{V}$, provides the strength (weight) of their connections. More generally, we refer to $\mathcal{A}$ as an affinity matrix, in which each entry $\mathcal{A}_{i,j}$ gives an affinity score between $\mathcal{V}_i$ and $\mathcal{V}_j$. In addition to the affinity matrix, which describes the topology of the graph and the relationships between nodes, it is common practice to define a signal $\mathcal{F} : \mathcal{V} \to \mathbb{R}^{d_{IN}}$ on the vertex set, which associates a d_{IN} dimensional feature vector to each node of the graph.

Graph Soft Clustering Given $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, we define a soft $\mathcal{K}$-partition of the graph a function that associates at each vertex $\mathcal{V}_i \in \mathcal{V}$ a membership value, in probabilistic terms, to each of the $|\mathcal{K}|$ cluster. The $\mathcal{K}$-partition can be shortly represented by a stochastic matrix $K \in \mathbb{R}^{\mathcal{N} \times |\mathcal{K}|}$ where the element $K_{i,k}$ equals the probability of vertex $\mathcal{V}_i$ belonging to cluster $\mathcal{K}_k$, $\mathrm{P}(\mathcal{V}_i \in \mathcal{K}_k)$. Given the affinity matrix $\mathcal{A} \in \mathbb{R}^{\mathcal{N} \times \mathcal{N}}$, we compute the following matrix:

$$\mathcal{A}^{\mathcal{K}} = K^{\mathrm{T}}(\mathcal{A} - I_{\mathcal{N}} \odot \mathcal{A})K, \tag{2.1}$$

where $I_{\mathcal{N}}$ indicates the identity matrix of size $\mathcal{N}$ [1]. $\mathcal{A}^{\mathcal{K}} \in \mathbb{R}^{|\mathcal{K}|\times|\mathcal{K}|}$ is highly related to the affinity matrix of the graph that can be obtained by applying to the original graph the soft $\mathcal{K}$-partition described by K. Indeed, if all the membership distributions behaved like a multivariate Kronecker Delta distribution, given an adjacency matrix $\mathcal{A}$ describing an undirected graph, $\mathcal{A}^{\mathcal{K}}_{k,k}$ $k = 1, 2, \ldots, |\mathcal{K}|$ would be equal to the double of the number of edges existing between the nodes insides the k-th cluster, and $\mathcal{A}^{\mathcal{K}}_{k,k'}$ $k, k' = 1, 2, \ldots, |\mathcal{K}|$ $k \neq k'$ would be equal to the number of edges connecting pair of nodes respectively belonging to the k-th and k'-th cluster. Likewise, in the soft case, we have:

$$\begin{aligned} \mathcal{A}^{\mathcal{K}}_{k,k} &= \textbf{\textit{Cohesion}}\,(K_k) = 2 \sum_{(\mathcal{V}_i,\mathcal{V}_j)\in\binom{\mathcal{V}}{2}} K_{i,k}K_{j,k}\,\mathcal{A}_{i,j}, \\ \mathcal{A}^{\mathcal{K}}_{k,k'} &= \sum_{i=1}^{\mathcal{N}} K_{i,k} \sum_{\substack{j=1 \\ j\neq i}}^{\mathcal{N}} K_{j,k'}\,\mathcal{A}_{i,j}. \end{aligned} \tag{2.2}$$

In such form, $\mathcal{A}^{\mathcal{K}}_{k,k'}$ can be considered an affinity measure between the k-th and k'-th nodes in the graph reduced by K. We consider as a 'good' soft $\mathcal{K}$-partition a partition that produces cluster with maximal cohesion. However, equivalent to ratio and normalized cut [199], we penalise imbalanced solutions through the addition of a penalty related to the size of each cluster:

$$\begin{aligned} \max_{K\in\mathbb{R}^{\mathcal{N}\times|\mathcal{K}|}} C(K) \quad &= \frac{1}{2}\sum_{k=1}^{|\mathcal{K}|} \frac{\textit{Cohesion}\,(K_k)}{\textit{Vol}\,(K_k)} \\ &= \frac{1}{2}\mathbf{1}^{\mathrm{T}}_{|\mathcal{K}|}\left[\operatorname{diag}(\mathcal{A}^{\mathcal{K}}) \oslash (K^{\mathrm{T}}D)\right] \\ \text{subject to} \quad &\sum_{k=1}^{|\mathcal{K}|} K_{i,k} = 1 \quad i = 1, 2, \ldots, \mathcal{N}, \\ \text{where} \quad \textbf{\textit{Vol}}\,(K_k) &= \sum_{i=1}^{\mathcal{N}} D_i\,\mathrm{P}(\mathcal{V}_i \in \mathcal{K}_k) \quad k = 1, 2, \ldots, |\mathcal{K}|. \end{aligned} \tag{2.3}$$

and $\oslash$ indicates the entry-wise division between two vectors of the same length, and $D \in \mathbb{R}^{\mathcal{N}}$ stand for a column vector in which each entry is equal to the degree

[1] The subtraction of the diagonal is performed to avoid the consideration of self-connections during cluster affinity and *Cohesion* computations, in Eq. 2.2.

of the corresponding node. This way, we obtain clusters with maximal cohesion and, at the same time, minimum size. It is noted that the main difference between such formulation and the well-known normalized cut relies on the membership's definition, the latter being defined in our case by means of soft assignments.

Graph Hierarchical Soft Clustering Let consider $\mathcal{A}^{\mathcal{K}_1}$ as the affinity matrix of the graph that can be obtained by applying a soft $\mathcal{K}$-partition, given by $K^{(1)} \in \mathbb{R}^{|\mathcal{K}_0|\times|\mathcal{K}_1|}$, to the original graph described by $\mathcal{A}$, where $|\mathcal{K}_0| = \mathcal{N}$. We can now partition $\mathcal{A}^{\mathcal{K}_1}$, based on the entries of a generic matrix $K^{(2)} \in \mathbb{R}^{|\mathcal{K}_1|\times|\mathcal{K}_2|}$, in order to obtain a new affinity matrix $\mathcal{A}^{\mathcal{K}_2}$, and so on. More generally, a cascade of M soft-partitions, described by an ordered sequence of $\mathcal{A}^{\mathcal{K}_1}, \mathcal{A}^{\mathcal{K}_2}, \ldots, \mathcal{A}^{\mathcal{K}_\mathrm{M}}$, forms a soft dendrogram for the original graph. Thus, the problem of obtaining a good dendrogram, in which clusters at each level are characterized by maximal cohesion and minimum size, is formalised as follows:

$$\begin{aligned} \max_{\substack{K^{(i)}\in\mathbb{R}^{|\mathcal{K}_{i-1}|\times|\mathcal{K}_i|} \\ i=1,2,\ldots,\mathrm{M}}} \quad & \mathcal{L}_\mathcal{K} = \frac{1}{2}\sum_{m=1}^{\mathrm{M}}\sum_{k=1}^{|\mathcal{K}_m|}\frac{\textit{Cohesion}\,(K_k^{(m)})}{\textit{Vol}\,(K_k^{(m)})} \\ \text{subject to} \quad & \sum_{k=1}^{|\mathcal{K}_m|} K_{i,k}^{(m)} = 1 \quad \substack{i=1,2,\ldots,|\mathcal{K}_{m-1}| \\ m=1,2,\ldots,\mathrm{M}.} \end{aligned} \tag{2.4}$$

2.2 Proposed Approach

The purpose of our proposal is to exploit the clustering mechanism in order to define a convolutional-like operator, able to ensure equivariance to translation and weight sharing in graph contexts as standard convolutions do. At a high level, our CCP operator can be considered as a layer which, at step m, takes in input an affinity matrix $\mathcal{A}^{\mathcal{K}_m}$ and a multi-dimensional $\mathcal{F}^{(m)} \in \mathbb{R}^{|\mathcal{K}_m|\times d_{IN}}$ signal defined on the vertex set. The output is composed by a new reduced affinity matrix $\mathcal{A}^{\mathcal{K}_{m+1}}$ (reflecting the results of the cluster step) and a pooled signal $\mathcal{F}^{(m+1)} \in \mathbb{R}^{|\mathcal{K}_{m+1}|\times d_{OUT}}$ (reflecting the results of the filter step where d_{OUT} is the dimension of the newly computed features). All architectures used in our experiments are composed by stacking CCP layers, which combine the pooling and filtering stage and, at the same time, increase the number of feature maps, as suggested in [22]. The objective in Eq. 2.4 is consequently optimised by backprogating gradients.

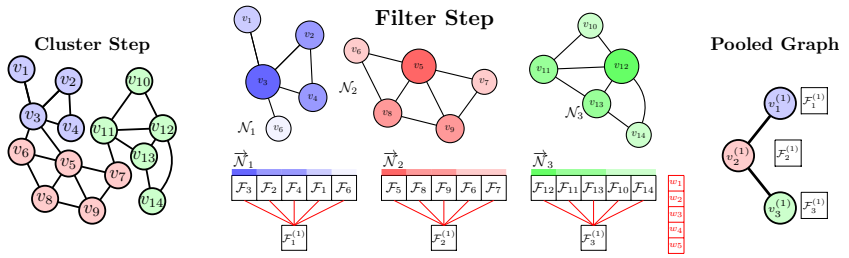


Figure 2.2: An illustration of the proposed CCP layer. Left, the cluster step outputs node's membership distribution among a pre-defined number of clusters. Centre, the filter step: a) selects, for each cluster, candidate nodes whose feature vectors will be aggregated; b) arranges such candidates according to a with-in cluster centrality score, building the support for the next step; c) aggregates feature vectors by means of a standard 1-d convolution, with stride equal to the kernel width (in this case, $L = 5$). Right, the result of CPP layer consists in a coarsened graph coupled with its filtered pooled signal. Best viewed in color.

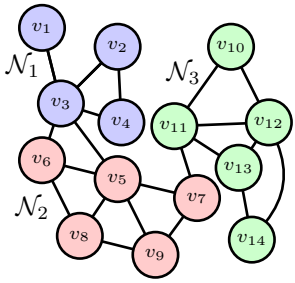
Cluster Step First of all, our model performs a soft-clustering step on the input graph (Fig. 2.2, left). To the purpose we define the stochastic matrix described in Section 2.1.1 as the output of a row-wise softmax applied on a variable matrix $U^{(m+1)} \in \mathbb{R}^{|\mathcal{K}_m| \times |\mathcal{K}_{m+1}|}$ learned during training:

$$K_{i,k}^{(m+1)} = P(\mathcal{V}_i^{(m)} \in \mathcal{K}_k^{(m+1)}) = \frac{e^{U_{i,k}^{(m+1)}}}{\sum_{k'=1}^{|\mathcal{K}_{m+1}|} e^{U_{i,k'}^{(m+1)}}} \tag{2.5}$$

where $i = 1, 2, \ldots, |\mathcal{K}_m|$ and $k = 1, 2, \ldots, |\mathcal{K}_{m+1}|$. In the second place, the downsampled affinity matrix $\mathcal{A}_{m+1}^{\mathcal{K}}$ describing the soft-partitioned graph induced by $K^{(m+1)}$ is computed by means of the quadratic form in Eq. 2.1. Eventually, we add a normalisation operation based on D [84] to prevent numerical instabilities:

$$\overline{\mathcal{A}}^{\mathcal{K}_{m+1}} = D^{-\frac{1}{2}} \mathcal{A}^{\mathcal{K}_{m+1}} D^{-\frac{1}{2}}. \tag{2.6}$$

Neighbourhood selection For each cluster $\mathcal{K}_k^{(m+1)}$, we select as candidate set $\mathcal{N}_k^{(m+1)}$ for the filtering stage the set containing the most L representative nodes



$\overrightarrow{\mathcal{N}_1}$	
$\mathcal{V}_i$	$Rank$
v_3	6
v_2	4
v_4	4
v_1	2
v_6	1

$\overrightarrow{\mathcal{N}}_2$	
$\mathcal{V}_i$	$Rank$
v_5	8
v_8	6
v_9	6
v_6	4
v_7	4

$\overrightarrow{\mathcal{N}}_3$	
$\mathcal{V}_i$	$Rank$
v_{12}	8
v_{11}	6
v_{13}	6
v_{10}	4
v_{14}	4

Figure 2.3: The ranking function (described by Equation 2.8) underpinning the filter step shown in Figure 2.2. The node colours denote cluster memberships. All edges have weight equal to one.

(where L is an hyperparameter) as:

$$\mathcal{N}_k^{(m+1)} = \underset{\mathcal{V}' \subset \mathcal{V}^{(m)}, |\mathcal{V}'|=L}{\operatorname{argmax}} \sum_{v \in \mathcal{V}'} Rank\,(v \to \mathcal{K}_k^{(m+1)}), \tag{2.7}$$

where the rank of a vertex $\mathcal{V}_i^{(m)}$ for a particular cluster $\mathcal{K}_k^{(m+1)}$ is given by its centrality in that cluster:

$$Rank\,(\mathcal{V}_i^{(m)} \to \mathcal{K}_k^{(m+1)}) = (1 + K_{i,k}^{(m+1)}) \sum_{\substack{j=1 \\ j \neq i}}^{|\mathcal{K}_m|} \mathcal{A}_{i,j}^{\mathcal{K}_m} K_{j,k}^{(m+1)}. \tag{2.8}$$

Intuitively, we consider a node more central if it has a high membership value for the cluster under consideration and, at the same time, a large part of its direct neighbours nodes share the same cluster in the input graph (Fig. 2.2, centre top).

Further, for each cluster, we compute its features as a linear combination over the feature vectors of its inner nodes. In doing so, we want to exploit the weight sharing property across all neighbours, keeping the parameters' number under control. To this end, we create a coherent support across clusters, in terms of their inner topological structure. In this respect, we propose to sort candidates by their centrality within the neighbourhood and then apply the same kernel to all clusters.

$$\begin{aligned} \overrightarrow{\mathcal{N}}_k^{(m+1)} &= (\mathcal{F}_{\phi(1)}^{(m)},\ \mathcal{F}_{\phi(2)}^{(m)},\ \ldots,\ \mathcal{F}_{\phi(L)}^{(m)}), \\ \text{with } \overrightarrow{\mathcal{N}}_k^{(m+1)}(l,i) &= \mathcal{F}_{\phi(l),i}^{(m)} \quad \substack{l=1,2,\ldots,L \\ i=1,2,\ldots,d_{IN}}. \end{aligned} \tag{2.9}$$

In simpler terms, the ordered set $\overrightarrow{\mathcal{N}}_k^{(m+1)}$ is recovered by sorting the candidates set $\mathcal{N}_k^{(m+1)}$ according to the $Rank$ function. By doing so, the l-th weight of the kernel is always multiplied by the feature vector $\mathcal{F}_{\phi(l)}^{(m)}$ being owned by the l-th node of the neighbour (in terms of centrality), namely $\mathcal{V}_{\phi(l)}^{(m)}$. Fig. 2.3 shows an example of the neighbourhood selection step for a simple graph.

Neighbourhood Aggregation The problem we face when sorting nodes by cluster centrality and then applying the same kernel to all neighbours, is that, by doing so we do not take into account the irregularity of the neighbour's shapes. As a matter of fact, the risk of this solution consists in the equal treatment, for different clusters, of nodes indexed in the same position by the sorting stage, whilst exhibiting considerably different centrality values. In order to mitigate such risk, we once again use the centrality measure to implement a gating mechanism on feature vectors during the aggregation phase. The underlying idea is to make the filtering operation invariant to different neighbours and let the gating mechanism address different cluster's structures and shapes. Roughly speaking, before applying the filtering operation described above, we are giving the centrality scores in input to a generic smoothed function $\sigma : \mathbb{R} \to (0, 1)$ (*e.g.* the sigmoid function). Once this has been done, we perform a point-wise multiplication on the feature vectors of each candidate node. The desired effect of this operation is to attenuate information coming from distant or non-central nodes and, at the same time, preserve signals coming from nodes that reside in the inner part of the cluster. Lastly, our model computes the pooled feature vector as follows:

$$\mathcal{F}_{k,j}^{(m+1)} = \sum_{i=1}^{d_{IN}} \sum_{l=1}^{L} W_{l,i,j} \left(\sigma_{k,l} \cdot \overrightarrow{\mathcal{N}}_k^{(m+1)}(l, i)\right) + b_j, \tag{2.10}$$

where $W \in \mathbb{R}^{L \times d_{IN} \times d_{OUT}}$ and $b \in \mathbb{R}^{d_{OUT}}$ are learnable parameters of our CCP layer, whereas $\sigma_{k,l}$ refers to the gate's activation value computed at $Rank\left(\mathcal{V}_{\phi(l)}^{(m)} \to \mathcal{K}_k^{(m+1)}\right)$. As shown in Fig. 2.2 (centre bottom), this operation is equivalent to a 1-d convolution, enabling weight sharing across clusters.

Optimisation Given a particular task, we simply add to the task-specific loss $\mathcal{L}_0$ (e.g. a cross-entropy) a term based on quality of the multi-level clustering solutions (Eq. 2.4) provided during the training phase:

$$\mathcal{L} = \mathcal{L}_0 + \mathcal{L}_{\mathcal{K}}. \tag{2.11}$$

Experiment	Input	Architecture						#params
NTU RGB-D	$(2000, 6)$	$(512, 256)$ $L = 16$	$(128, 384)$ $L = 16$	$(32, 512)$ $L = 8$	$(8, 768)$ $L = 8$	$(1, 1024)$ $L = 8$	FC1024	$\sim 14 \cdot 10^6$
CIFAR-10	$(1024, 3)$	$(256, 256)$ $L = 16$	$(64, 384)$ $L = 16$	$(16, 512)$ $L = 8$	$(4, 768)$ $L = 8$	$(1, 1024)$ $L = 4$	FC1024	$\sim 10 \cdot 10^6$
20NEWS	$(10000, 1)$	$(2048, 128)$ $L = 16$	$(512, 192)$ $L = 16$	$(128, 256)$ $L = 8$	$(32, 384)$ $L = 8$	$(4, 512)$ $L = 8$	$(1, 512)$ $L = 4$	FC256	$\sim 25 \cdot 10^6$

Table 2.1: Summary of the architectures used in our experiments. We indicate with $(|\mathcal{K}|, d_{OUT})$ the number of nodes and feature maps of each layer. Note that a further softmax layer is employed to estimate class probabilities.

It is important to note that the presence of the supervision signal may provide information to the process of clusters formation, backpropagating its gradient towards all U variables (Eq. 2.5).

2.3 Experiments

In order to show the generality and effectiveness of our model for classification, we apply our architecture to three different domains. First, we train our model to classify human actions, given the 3D coordinates of each skeleton's joint: to this end, we evaluate it on NTU RGB-D dataset [171]. Secondly, we conduct experiments on image classification. More specifically, we use CIFAR-10 [88] as benchmark test, which is a challenging dataset for non-CNN architectures. Finally, we apply our solution on the 20NEWS dataset, where the goal is to address a text categorisation problem.

Implementation details In each experiment, all parameters are learned using Adam [82] as an optimisation algorithm, with an initial learning rate fixed to 0.001. We use ELU [32] as activation function and Batch Normalization [68] in all layers to speed up the convergence. Moreover, we apply dropout and l_2 weight regularisation (with value 10^{-4}) to prevent overfitting, as well as standard data augmentation for CIFAR-10 and noise injection coupled with random 3d rotations for NTU RGB-D. All the others architectures' hyperparameters are summarised in Tab. 2.3. In each experiment, we subsample the input graph until its cardinality becomes equal to one: afterwards, we feed its feature vector into two fully connected layers, followed by a softmax layer providing the target class predictions.

Method	Cross Subject	Cross View
Lie Group [195]	50.1	52.8
HBRNN-L [44]	59.1	64.0
P-LSTM [171]	62.9	70.3
ST-LSTM+TS [108]	69.2	77.7
TGCNN [218]	71.4	82.9
Temporal Conv [81]	74.3	83.1
Deep STGC$_K$ [97]	74.9	86.3
C-CNN + MTLN [77]	79.6	84.8
CCP (our)	**80.1**	**86.8**

Table 2.2: Summary of results in terms of classification accuracy for NTU RGB+D.

2.3.1 Experimental Results

Action Recognition The NTU RGB+D Human Activity Dataset [171] is one of the largest datasets for human action recognition. It contains 56,880 action samples for 60 different actions, captured by the Kinect v.2 sensors. Each sample, showing a daily action performed by one or two participants, is made available in 4 different modalities: RGB videos, depth map sequences, 3D skeletal data and infrared videos. Since we are interested in graph classification, in order to perform action recognition, we just use the 3D skeletal data, represented by a temporal sequence of 25 joints. To this end, we model each sequence as a signal $\mathcal{F}^{(0)} \in \mathbb{R}^{(25 \cdot T) \times 3}$ defined on a single fixed spatio temporal graph, whose structure can be summarised as follows: a vertex set $\mathcal{V}_{ST} = \{v_{i,t} | i = 1, 2, \dots, 25\ , t = 1, 2, \dots, T\}$, which includes all joints captured in a fixed length sequence ($T = 80$). The edge set $\mathcal{E}_{ST}$ can be defined as the union of two distinct subsets: $\mathcal{E}_S$, which contains all edges within each frame according to the natural human-body connectivity, and $\mathcal{E}_T$, which includes all edges existing between the same joint in two adjacent frame. In order to evaluate the model's performance, we run two different standard benchmarks as in [171]: the cross-subject setting, in which the train/test split is based on two disjoint sets of actors; and the cross-view setting, where the test samples are captured from a different camera from those collecting the training sequences. Tab. 2.2 reports the classification accuracy on both settings, comparing it with other approaches: as can be appreciated, CCP outperforms previous state-of-the-art methods, (including graph-oriented architectures [218, 97]) despite being more general and not designed to only address action recognition settings.

Method	Accuracy
Graph-CNNs [176]	68.3
FC [105]	78.6
CCP (our)	**84.4**
Stochastic Pooling [211]	84.9
ResNet [58]	93.6

Table 2.3: Image classification accuracy on CIFAR-10.

Method	Accuracy
Linear SVM †	65.9
Softmax †	66.3
Multinomial Naive Bayes †	68.5
FC2500-FC500 †	65.8
Chebyshev - GC32 [36] †	68.3
CCP (our)	**70.1**

† Baselines' results published in [36].

Table 2.4: Text categorisation accuracy on 20NEWS.

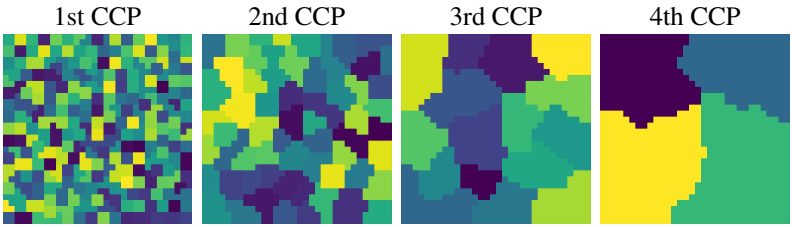


Figure 2.4: Receptive fields learned during CIFAR10 training.

Image Classification We conduct experiments on CIFAR-10, a popular dataset widely used for image recognition. Each image, labeled into one of ten classes, can be treated as a signal defined on a graph, which can in turn be modeled as a 32×32 grid structure. In particular, every pixel is a vertex such a graph, linked to its neighbours following a 8-connectivity. The colour information is encoded as a signal $\mathcal{F}^{(0)} \in \mathbb{R}^{1024\times 3}$ over such vertexes. As shown in Tab. 2.3, CCP obtains an encouraging performance in terms of classification accuracy on test set. Indeed, our method outperforms both the best reported fully connected (FC) network [105] and Graph-CNNs [176] - to the best of our knowledge, the only graph classification model in literature that reports results on CIFAR-10 - by a significant margin. To put our results into perspective, we report the performance obtained by [211], which is the nearest score founded in the literature given by a deep CNN, as well as the results of a state-of-art CNNs like [58]. The gap with respect to the latter is still consistent, suggesting that there is still room for improvement in euclidean domains. Fig. 2.4 also depicts an illustration of the hierarchical clustering computed on the input grid. As can be seen, as

Filter	Coarsen	GAP	CIFAR	NTU-CS
Chebyshev [36]	Graclus	–	78.15	74.85
GCN [84]	Graclus	–	67.01	62.00
GAT	Graclus	–	72.82	59.48
GAT	-	✓	66.39	26.74 †
CCP (ours)	CCP	–	**84.4**	**80.1**

† We found it extremely hard to train due to the huge memory footprint required.

Table 2.5: Comparison of different graph corsening and filtering approches on CIFAR-10 and Cross Subject NTU RGB+D.

the input image undergoes CCP layers, its representations are computed out of compact regions, resembling dyadic clustering that has been proven a successful downsampling strategy in CNNs.

Text Categorisation In order to further validate the quality of our proposal in diverse data domains, we apply our model on text categorisation. In this respect, we conduct experiments on the 20NEWS dataset [76], adhering to the guidelines described in [36] for the construction of the shared graph. To summarise, such protocol models each text as a graph which has a node for each common word in the document set. On the other hand, the pairwise connectivities of such graph are shared and obtained assessing the similarities inducted by word2vec embeddings [131], followed by a discretisation step computed through a K-NN pass (with $K = 16$). This way, each document $\mathcal{D}$ can be represented as a signal over a fixed graph, implemented as the word's distribution observed in $\mathcal{D}$. As indicated in Tab. 2.4, the discussed approach leads to good performances, defeating both baselines and the graph convolutional layer based on polynomial spectral filters. On this latter point, our architecture seems to take advantages of its depth and hierarchical nature, differently from [36] where a shallow graph convolutional network has been employed to categorise documents.

2.4 Model Analysis

Comparisons with other coarsening approaches We further compare our proposal w.r.t. three different works (Tab. 2.5): GCN [84], Chebyshev filtering [36]

Loss $\mathcal{L}$	Gradient	CIFAR-10	NTU-CS
$\mathcal{L}_0 + \mathcal{L}_\mathcal{K}$ (Eq. 2.11)	-	**84.4**	**80.1**
$\mathcal{L}_0$	-	66.7	73.1
$\mathcal{L}_0$	$\delta\mathcal{L}_0/\delta U \leftarrow 0$	56.8	70.6
$\mathcal{L}_0 + \mathcal{L}_\mathcal{K}$	$\delta\mathcal{L}_0/\delta U \leftarrow 0$	83.8	78.6

Table 2.6: Ablative results under different optimisations.

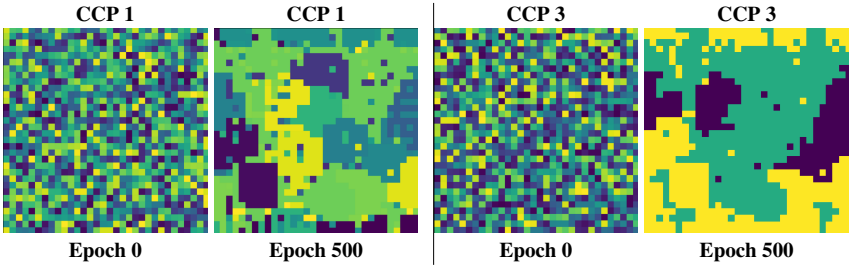


Figure 2.5: Receptive fields arising from $\mathcal{L}_0$ minimisation on CIFAR-10.

and Graph Attention Networks (GAT) [194]. In this respect we use the Graclus algorithm [158] for coarsening the input graph and vary the graph filtering strategy accordingly to the referenced work. For all the experiments we keep architectural settings described in Tab. 2.3 and use the public implementation of these works. Futhermore, we also design a non-coarsening baseline by performing a global average pooling (GAP) on nodes features (after GAT manipulation) before the fully connected classification layers. The experiment suggests that CCP outperforms, by a consistent margin, the previous GCN+coarsening approach. Moreover, it still outperforms Graclus as a coarsening strategy even though recent filters such as GAT are applied. In this regard, we empirically observed that order-invariant filters (*e.g.* GAT), despite being more general, may treat the same graph differently, according to the attention scores. This is in fact a great advantage when graph layout may vary across examples, though potentially unrewarding when the support remains the same through all the dataset.

The impact of the task-specific loss We studied the contribution of the loss $\mathcal{L}_0$ (Eq. 2.11) and found three evidences supporting its beneficial effect: *i)* if

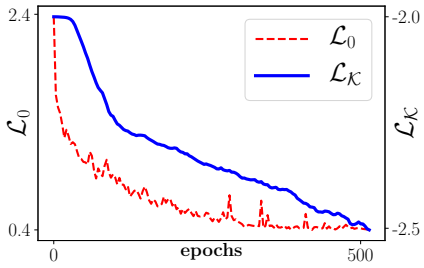
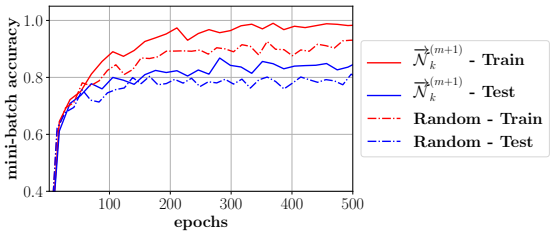


Figure 2.6: Loss landscapes under $\mathcal{L}_0$ minimisation.



Order	Accuracy
Cluster Centrality	84.4
Random	80.3

Figure 2.7: Results from the ablation study conducted on CIFAR-10. The top picture shows test and training learning curve under both settings.

only $\mathcal{L}_0$ is optimised, then suppressing its gradients on membership variables (*i.e.* cluster memberships are randomly fixed and cannot be changed during training) leads to poorer performances ($\mathcal{L}_0, \delta\mathcal{L}_0/\delta U \leftarrow 0$ against $\mathcal{L}_0$, Tab. 2.6); *ii)* when both objectives are optimised, discarding gradients of $\mathcal{L}_0$ on membership variables yields slightly degraded results ($\mathcal{L}_0+\mathcal{L}_\mathcal{K}, \delta\mathcal{L}_0/\delta U \leftarrow 0$, against $\mathcal{L}_0+\mathcal{L}_\mathcal{K}$, Tab. 2.6); *iii)* even when only optimising $\mathcal{L}_0$ we can observe the emergence of compact regions (*i.e.* clusters) in the clustering landscape (Fig. 2.5). Another evidence of this effect is the lowering of the $\mathcal{L}_\mathcal{K}$ when the network is optimised w.r.t. $\mathcal{L}_0$ (Fig. 2.6).

Effectiveness of the Ranking function Finally, we conducted an ablation study for validating the effectiveness of the proposed within-cluster centrality measure in capturing shift-invariant structures on the graph. To this end, we compared it with a less principled criteria, involving a random permutation of the candidate

nodes. Specifically, under the random setting, we still keep the definition of neighbourhood $\mathcal{N}_k^{(m+1)}$ given by Eq. 2.7. However, instead of sorting nodes inside it, we randomly sample a fixed permutation for each cluster, before the start of the learning. Without the sorting criteria, learnable kernels cannot rely on a coherent topological structure within their support, weakening the effect of weight sharing. We evaluated both of the policies on CIFAR-10, and reported our results in Fig. 2.7, in terms of learning curves and test error. The figure suggests that the sorting criterion indeed leads to a significant improvement in performance, due to a proper exploitation of weight sharing.

2.5 Conclusion

In this chapter we have proposed a novel approach for graph signal classification, leveraging both local and global structures, the latters arising from a multi-scale and hierarchical representation of the input signal. The main contribution consists in a layer which performs a (soft) clustering step on the input graph and, accordingly, aggregates information within each cluster. Experiments show that our model consistently outperforms recent graph-based classification models in different data domains. The ablation study suggests that the proposed layer successfully exploit the weight sharing property in a graph convolutional architecture.

Chapter 3

Novelty Detection

Novelty detection is defined as the identification of samples exhibiting significantly different traits with respect to an underlying model of regularity, built from a collection of normal samples. The awareness of an autonomous system to recognize unknown events enables applications in several domains, ranging from video surveillance [24, 56] to defect detection [91]. Moreover, the surprise inducted by unseen events is emerging as a crucial aspect in reinforcement learning settings, as an enabling factor in curiosity-driven exploration [141].

However, in this setting, the definition and labeling of novel examples are not possible. Accordingly, the literature agrees on approximating the ideal shape of the boundary separating normal and novel samples by modeling the intrinsic characteristics of the former. Therefore, prior works tackle such problem by following principles derived from the unsupervised learning paradigm [34, 160, 56, 111, 122]. Due to the lack of a supervision signal, the process of feature extraction and the rule for their normality assessment can only be guided by a proxy objective, assuming the latter will define an appropriate boundary for the application at hand.

According to cognitive psychology [12], novelty can be expressed either in terms of capabilities to *remember* an event or as a degree of *surprisal* [185] aroused by its observation. The latter is mathematically modeled in terms of low probability to occur under an expected model, or by lowering a variational free energy [71]. In this framework, prior models take advantage of either parametric [227] or non-parametric [61] density estimators. Differently, remembering an event implies the adoption of a memory represented either by a dictionary of normal prototypes

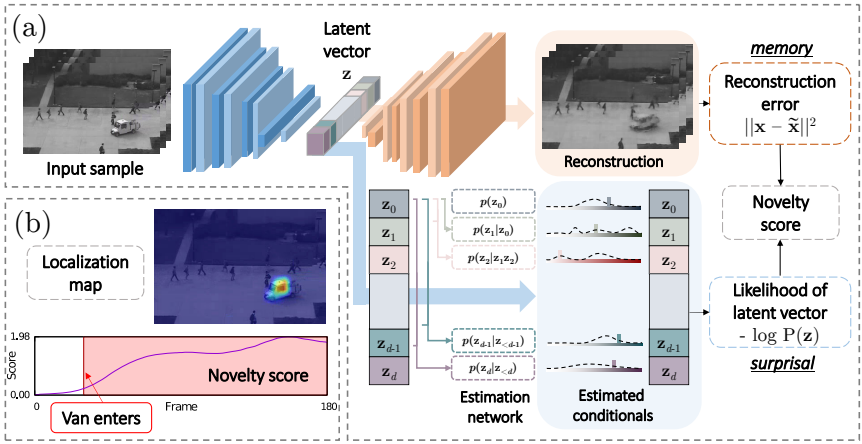


Figure 3.1: The proposed framework. The overall architecture, depicted in (a), consists of a deep autoencoder and an autoregressive estimation network operating on its latent space. In (b), the joint minimization of their respective objective leads to a measure of novelty obtained by assessing the remembrance of the model when looking to a new sample, combined with its surprise aroused by causal factors.

- as in sparse coding approaches [34] - or by a low dimensional representation of the input space, as in the self-organizing maps [86] or, more recently, in deep autoencoders. Thus, in novelty detection, the remembering capability for a given sample is evaluated either by measuring reconstruction errors [56, 111] or by performing discriminative in-distribution tests [160]. Our proposal contributes to the field by merging remembering and surprisal aspects into a unique framework: we design a generative unsupervised model (*i.e.*, an autoencoder, represented in Fig. 3) that exploits end-to-end training in order to maximize remembering effectiveness for normal samples whilst minimizing the surprisal of their latent representation. This latter point is enabled by the maximization of the likelihood of latent representations through an autoregressive density estimator, which is performed in conjunction with the reconstruction error minimization. We show that, by optimizing both terms jointly, the model implicitly seeks for minimum entropy representations maintaining its remembering/reconstructive power. While entropy minimization approaches have been adopted in deep neural compression [10], to our knowledge this is the first proposal tailored for novelty detection. In memory

terms, our procedure resembles the concept of prototyping the normality using as few templates as possible. Moreover, evaluating the output of the estimator enables the assessment of the surprisal aroused by a given sample.

3.1 Preliminaries

Reconstruction-based methods On the one hand, many works lean toward learning a parametric projection and reconstruction of normal data, assuming outliers will yield higher residuals. Traditional sparse-coding algorithms [220, 34, 118] adhere to such framework, and represent normal patterns as a linear combination of a few basis components, under the hypotheses that novel examples would exhibit a non-sparse representation in the learned subspace. In recent works, the projection step is typically drawn from deep autoencoders [56]. In [122] the authors recover sparse coding principles by imposing a sparsity regularization over the learned representations, while a recurrent neural network enforces their smoothness along the time dimension. In [160], instead, the authors take advantage of an adversarial framework in which a discriminator network is employed as the actual novelty detector, spotting anomalies by performing a discrete in-distribution test. Oppositely, future frame prediction [111] maximizes the expectation of the next frame exploiting its knowledge of the past ones; at test time, observed deviations against the predicted content advise for abnormality. Differently from the above-mentioned works, the proposal discussed in this chapter relies on modeling the prior distribution of latent representations. This choice is coherent with recent works from the density estimation community [183, 14]. However, to the best of our knowledge, our work is the first advocating for the importance of such a design choice for novelty detection.

Probabilistic methods A complementary line of research investigates different strategies to approximate the density function of normal appearance and motion features. The primary issue raising in this field concerns how to estimate such densities in a high-dimensional and complex feature space. In this respect, prior works involve hand-crafted features such as optical flow or trajectory analysis and, on top of that, employ both non-parametric [2] and parametric [13, 124, 99] estimators, as well as graphical modeling [80, 92]. Modern approaches rely on deep representations (*e.g.*, captured by autoencoders), as in Gaussian classifiers [159] and Gaussian Mixtures [227]. In [61] the authors involve a Kernel Density Estimator (KDE) modeling activations from an auxiliary object detection network. A

recent research trend considers training Generative Adversarial Networks (GANs) on normal samples. However, as such models approximate an implicit density function, they can be queried for new samples but not for likelihood values. Therefore, GAN-based models employ different heuristics for the evaluation of novelty. For instance, in [163] a guided latent space search is exploited to infer it, whereas [148] directly queries the discriminator for a normality score.

3.2 Proposed Approach

Maximizing the probability of latent representations is analogous to lowering the surprisal for a normal configuration, defined as the negative log-density of a latent variable instance [185]. Conversely, remembering capabilities can be evaluated by the reconstruction accuracy of a sample under its latent representation.

We model these aspects in a latent variable model setting, where the density function of training samples $p(\mathbf{x})$ is modeled through an auxiliary random variable $\mathbf{z}$, describing the set of causal factors underlying all observations. By factorizing

$$p(\mathbf{x}) = \int p(\mathbf{x}|\mathbf{z})p(\mathbf{z})d\mathbf{z}, \tag{3.1}$$

where $p(\mathbf{x}|\mathbf{z})$ is the conditional likelihood of the observation given a latent representation $\mathbf{z}$ with prior distribution $p(\mathbf{z})$, we can explicit both the memory and surprisal contribution to novelty. We approximate the marginalization by means of an inference model responsible for the identification of latent space vector for which the contribution of $p(\mathbf{x}|\mathbf{z})$ is maximal. Formally, we employ a deep autoencoder, in which the reconstruction error plays the role of the negative logarithm of $p(\mathbf{x}|\mathbf{z})$, under the hypothesis that $p(\mathbf{x}|\mathbf{z}) = \mathcal{N}(\mathbf{x}|\tilde{\mathbf{x}}, I)$ where $\tilde{\mathbf{x}}$ denotes the reconstruction. Additionally, surprisal is incorporate by equipping the autoencoder with an auxiliary deep parametric estimator learning the prior distribution $p(\mathbf{z})$ of latent vectors, and training it by means of Maximum Likelihood Estimation (MLE). Our architecture is therefore composed of three building blocks (Fig. 3.2): an encoder $f(\mathbf{x};\theta_f)$, a decoder $g(\mathbf{z};\theta_g)$ and a probabilistic model $h(\mathbf{z};\theta_h)$:

$$\begin{gathered} f(\mathbf{x};\theta_f) : \mathbb{R}^m \to \mathbb{R}^d, \qquad g(\mathbf{z};\theta_g) : \mathbb{R}^d \to \mathbb{R}^m, \\ h(\mathbf{z};\theta_h) : \mathbb{R}^d \to [0,1]. \end{gathered} \tag{3.2}$$

The encoder processes input $\mathbf{x}$ and maps it into a compressed representation $\mathbf{z} = f(\mathbf{x};\theta_f)$, whereas the decoder provides a reconstructed version of the input

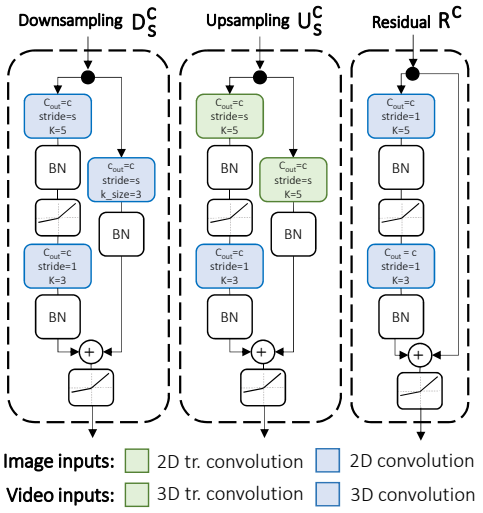


Figure 3.2: Building blocks employed in the autoencoder's architecture.

$\tilde{\mathbf{x}} = g(\mathbf{z}; \theta_g)$. The probabilistic model $h(\mathbf{z}; \theta_h)$ estimates the density in $\mathbf{z}$ via an autoregressive process, allowing to avoid the adoption of a specific family of distributions (*i.e.*, Gaussian), potentially unrewarding for the task at hand.

With such modules we can assess the two sources of novelty: elements whose observation is poorly explained by the causal factors inducted by normal samples (*i.e.*, high reconstruction error); elements exhibiting good reconstructions whilst showing surprising underlying representations under the learned prior.

Autoregressive density estimation Autoregressive models provide a general formulation for tasks involving sequential predictions, in which each output depends on previous observations [119, 193]. We adopt such a technique to factorize a joint distribution, thus avoiding to define its landscape a priori [93, 187]. Formally, $p(\mathbf{z})$ is factorized as

$$p(\mathbf{z}) = \prod_{i=1}^{d} p(z_i | \mathbf{z}_{<i}), \tag{3.3}$$

so that estimating $p(\mathbf{z})$ reduces to the estimation of each single Conditional Probability Density (CPD) expressed as $p(z_i|\mathbf{z}_{<i})$, where the symbol $<$ implies an order

over random variables. Some prior models obey handcrafted orderings [192, 191], whereas others rely on order agnostic training [188, 50]. Nevertheless, it is still not clear how to estimate the proper order for a given set of variables. In our model, this issue is directly tackled by the optimization.

From a technical perspective, the estimator $h(\mathbf{z};\theta_h)$ outputs parameters for d distributions $p(z_i|\mathbf{z}_{<i})$. In our implementation, each CPD is modeled as a multinomial over B=100 quantization bins. To ensure a conditional estimate of each underlying density, we design proper layers guaranteeing that the CPD of each symbol z_i is computed from inputs $\{z_1,\ldots,z_{i-1}\}$ only.

Objective and connection with differential entropy The three components f, g and h are jointly trained to minimize $\mathcal{L} \equiv \mathcal{L}(\theta_f,\theta_g,\theta_h)$ as follows:

$$\begin{aligned}\mathcal{L} &= \mathcal{L}_{\text{REC}}(\theta_f,\theta_g) + \lambda\mathcal{L}_{\text{LLK}}(\theta_f,\theta_h)\\ &= \mathbb{E}_{\mathbf{x}}\Big[\underbrace{||\mathbf{x}-\tilde{\mathbf{x}}||^2}_{\text{reconstruction term}} - \lambda\underbrace{\log(h(\mathbf{z};\theta_h))}_{\text{log-likelihood term}}\Big],\end{aligned} \tag{3.4}$$

where λ is a hyper-parameter controlling the weight of the $\mathcal{L}_{\text{LLK}}$ term. It is worth noting that it is possible to express the log-likelihood term as

$$\begin{aligned}&\mathbb{E}_{\mathbf{z}\sim p^*(\mathbf{z};\theta_f)}\big[-\log h(\mathbf{z};\theta_h)\big]\\ &= \mathbb{E}_{\mathbf{z}\sim p^*(\mathbf{z};\theta_f)}\big[-\log h(\mathbf{z};\theta_h) + \log p^*(\mathbf{z};\theta_f) - \log p^*(\mathbf{z};\theta_f)\big]\\ &= D_{\text{KL}}(p^*(\mathbf{z};\theta_f)\parallel h(\mathbf{z};\theta_h)) + \mathbb{H}[p^*(\mathbf{z};\theta_f)],\end{aligned} \tag{3.5}$$

where $p^*(\mathbf{z};\theta_f)$ denotes the true distribution of the codes produced by the encoder, and is therefore parametrized by θ_f. This reformulation of the MLE objective yields meaningful insights about the entities involved in the optimization. On the one hand, the Kullback-Leibler divergence ensures that the information gap between our parametric model h and the true distribution p^* is small. On the other hand, this framework leads to the minimization of the differential entropy of the distribution underlying the codes produced by the encoder f. Such constraint constitutes a crucial point when learning normality. Intuitively, if we think about the encoder as a source emitting symbols (namely, the latent representations), its desired behavior, when modeling normal aspects in the data, should converge to a 'boring' process characterized by an intrinsic low entropy, since surprising and novel events are unlikely to arise during the training phase. Accordingly, among all the possible settings of the hidden representations, the objective begs the encoder

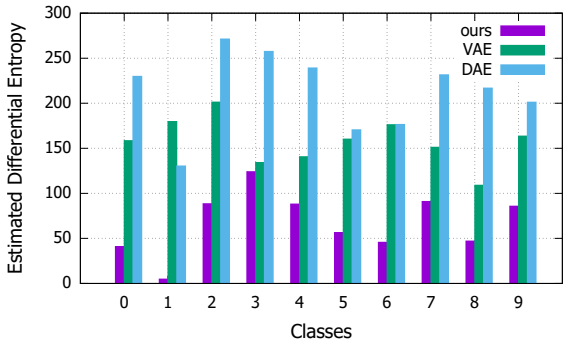


Figure 3.3: Estimated differential entropies delivered on each MNIST class in the presence of different regularization strategies: our, divergence w.r.t a Gaussian prior (VAE) and input perturbation (DAE). For each class, the estimate is computed on the training samples' hidden representations, whose distribution are fit utilizing a Gaussian KDE in a 3D-space. All models being equal, ours exhibits lower entropies on all classes.

to exhibit a low differential entropy, leading to the extraction of features that are easily predictable, therefore common and recurrent within the training set. These features are indeed the most useful to distinguish novel samples from the normal ones, making our proposal a suitable regularizer in the anomaly detection setting.

We report empirical evidence of the decreasing differential entropy in Fig. 3.3 comparing the behavior of the same model under different regularization strategies.

3.2.1 Architectural Components

Autoencoder blocks Encoder and decoder are respectively composed by down-sampling and upsampling residual blocks depicted in Fig. 3.2. The encoder ends with fully connected (FC) layers. When dealing with video inputs, we employ *causal* 3D convolutions [8] within the encoder (*i.e.*, only accessing information from previous time-steps). Moreover, at the end of the encoder, we employ a temporally-shared full connection (TFC, namely a linear projection sharing parameters across the time axis on the input feature maps) resulting in a temporal series of feature vectors. This way, the encoding procedure does not shuffle information across time-steps, ensuring temporal ordering.

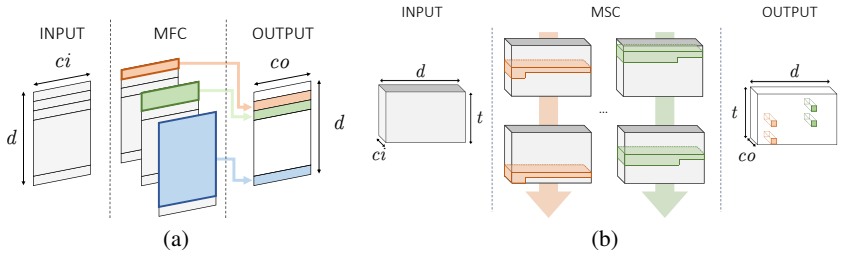


Figure 3.4: Proposed autoregressive layers, namely the Masked Fully Connection (a, Eq. 3.6) and the Masked Stacked Convolution (b, Eq. 3.7). For both layers, we represent type A structure. Different kernel colors represent different parametrizations.

Autoregressive layers To guarantee the autoregressive nature of each output CPD, we need to ensure proper connectivity patterns in each layer of the estimator h. Moreover, since latent representations exhibit different shapes depending on the input nature (image or video), we propose two different solutions.

When dealing with images, the encoder provides feature vectors with dimensionality d. The autoregressive estimator is composed by stacking multiple Masked Fully Connections (MFC, Fig. 3.4-(a)). Formally, it computes output feature map $\mathbf{o} \in \mathbb{R}^{d \times co}$ (where co is the number of output channels) given the input $\mathbf{h} \in \mathbb{R}^{d \times ci}$ (assuming $ci = 1$ at the input layer). The connection between the input element $\mathbf{h}_i^k$ in position i, channel k and the output element $\mathbf{o}_j^l$ is parametrized by

$$\begin{cases} w_{i,j}^{k,l} & \text{if } i < j \\ \begin{cases} w_{i,j}^{k,l} & \text{if type = B} \\ 0 & \text{if type = A} \end{cases} & \text{if } i = j \\ 0 & \text{if } i > j. \end{cases} \tag{3.6}$$

Type A forces a strict dependence on previous elements (and is employed only as the first estimator layer), whereas type B masks only succeeding elements. Assuming each CPD modeled as a multinomial, the output of the last autoregressive layer (in $\mathbb{R}^{d \times B}$) provides probability estimates for the B bins that compose the space quantization.

On the other hand, the compressed representation of video clips has dimensionality $t \times d$, being t the number of temporal time-steps and d the length of the

code. Accordingly, the estimation network is designed to capture two-dimensional patterns within observed elements of the code. However, naively plugging 2D convolutional layers would assume translation invariance on both axes of the input map, whereas, due to the way the compressed representation is built, this assumption is only correct along the temporal axis. To cope with this, we apply d different convolutional kernels along the code axis, allowing the observation of the whole feature vector in the previous time-step as well as a portion of the current one. Every convolution is free to stride along the time axis and captures temporal patterns. In such operation, named Masked Stacked Convolution (MSC, Fig. 3.4-(b)), the i-th convolution is equipped with a kernel $\mathbf{w}^{(i)} \in \mathbb{R}^{3\times d}$ kernel, that gets multiplied by the binary mask $\mathbf{M}^{(i)}$, defined as

$$m_{j,k}^{(i)} \in \mathbf{M}^{(i)} = \begin{cases} 1 & \text{if } j = 0 \\ 1 & \text{if } j = 1 \text{ and } k < i \text{ and type=A} \\ 1 & \text{if } j = 1 \text{ and } k \leq i \text{ and type=B} \\ 0 & \text{otherwise,} \end{cases} \tag{3.7}$$

where j indexes the temporal axis and k the code axis.

Every single convolution yields a column vector, as a result of its stride along time. The set of column vectors resulting from the application of the d convolutions to the input tensor $\mathbf{h} \in \mathbb{R}^{t\times d\times ci}$ are horizontally stacked to build the output tensor $\mathbf{o} \in \mathbb{R}^{t\times d\times co}$, as follows:

$$\mathbf{o} = \Big|\Big|_{i=1}^{d} [(\mathbf{M}^{(i)} \odot \mathbf{w}^{(i)}) * \mathbf{h}], \tag{3.8}$$

where $||$ represents the horizontal concatenation operation.

3.3 Experiments

We test our solution in three different settings: images, videos, and cognitive data. In all experiments the novelty assessment on the i-th example is carried out by summing the reconstruction term (REC_i) and the log-likelihood term (LLK_i) in Eq. 3.4 in a single novelty score NS_i:

$$NS_i = norm_S(REC_i) + norm_S(LLK_i). \tag{3.9}$$

	MNIST						
	OC SVM	**KDE**	**DAE**	**VAE**	**Pix CNN**	**GAN**	**ours**
0	0.988	0.885	0.991	0.998	0.531	0.926	0.993
1	0.999	0.996	0.999	0.999	0.995	0.995	0.999
2	0.902	0.710	0.891	0.962	0.476	0.805	0.959
3	0.950	0.693	0.935	0.947	0.517	0.818	0.966
4	0.955	0.844	0.921	0.965	0.739	0.823	0.956
5	0.968	0.776	0.937	0.963	0.542	0.803	0.964
6	0.978	0.861	0.981	0.995	0.592	0.890	0.994
7	0.965	0.884	0.964	0.974	0.789	0.898	0.980
8	0.853	0.669	0.841	0.905	0.340	0.817	0.953
9	0.955	0.825	0.960	0.978	0.662	0.887	0.981
avg	0.951	0.814	0.942	0.969	0.618	0.866	**0.975**

Table 3.1: AUROC results for novelty detection on MNIST. Each row represents a different class on which baselines and our model are trained.

Individual scores are normalized using a reference set of examples S (different for every experiment),

$$norm_S(L_i) = \frac{L_i - \min_{j \in S} L_j}{\max_{j \in S} L_j - \min_{j \in S} L_j}. \tag{3.10}$$

3.3.1 Image Novelty Detection

To assess the model's performance in one class settings, we train it on each class of either MNIST or CIFAR-10 separately. In the test phase, we present the corresponding test set, which is composed of 10000 examples of all classes, and expect our model to assign a lower novelty score to images sharing the label with training samples. We use standard train/test splits, and isolate 10% of training samples for validation purposes, and employ it as the normalization set (S in Eq. 3.9) for the computation of the novelty score.

As for the baselines, we consider the following:

- standard methods such as OC-SVM [165] and Kernel Density Estimator (KDE), employed out of features extracted by PCA-whitening;

	CIFAR10						
	OC SVM	**KDE**	**DAE**	**VAE**	**Pix CNN**	**GAN**	**ours**
0	0.630	0.658	0.718	0.688	0.788	0.708	0.735
1	0.440	0.520	0.401	0.403	0.428	0.458	0.580
2	0.649	0.657	0.685	0.679	0.617	0.664	0.690
3	0.487	0.497	0.556	0.528	0.574	0.510	0.542
4	0.735	0.727	0.740	0.748	0.511	0.722	0.761
5	0.500	0.496	0.547	0.519	0.571	0.505	0.546
6	0.725	0.758	0.642	0.695	0.422	0.707	0.751
7	0.533	0.564	0.497	0.500	0.454	0.471	0.535
8	0.649	0.680	0.724	0.700	0.715	0.713	0.717
9	0.508	0.540	0.389	0.398	0.426	0.458	0.548
avg	0.951	0.814	0.942	0.969	0.618	0.866	**0.975**

Table 3.2: AUROC results for novelty detection on CIFAR10.

- a denoising autoencoder (DAE) sharing the same architecture as our proposal, but defective of the density estimation module. The reconstruction error is employed as a measure of normality vs. novelty;
- a variational autoencoder (VAE) [83], also sharing the same architecture as our model, in which the Evidence Lower Bound (ELBO) is employed as the score;
- Pix-CNN [191], modeling the density by applying autoregression directly in the image space;
- the GAN-based approach illustrated in [163].

We report the comparison in Tab. 3.1 and Tab. 3.2: as can be seen, results are reported in terms ofthe Area Under Receiver Operating Characteristic (AUROC), which is the standard metric for the task. As the table shows, our proposal outperforms all baselines in both settings.

Considering MNIST, most methods perform favorably. Notably, Pix-CNN fails in modeling distributions for all digits but one, possibly due to the complexity of modeling densities directly on pixel space and following a fixed autoregression order. Such poor test performance are registered despite good quality samples that we observed during training: indeed, the weak correlation between sample quality

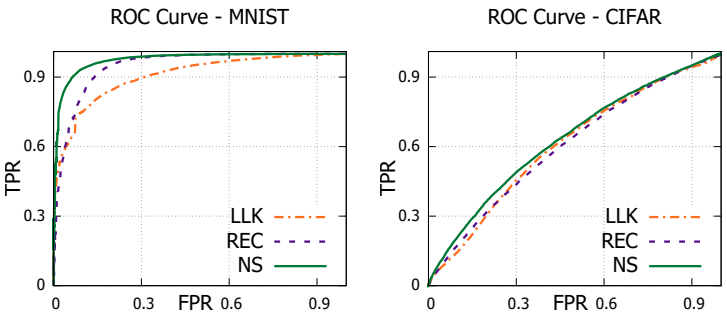


Figure 3.5: ROC curves delivered by different scoring strategies on MNIST and CIFAR-10 test sets. Each curve is an interpolation over the ten classes.

and test log-likelihood of the model has been motivated in [181]. Surprisingly, OC-SVM outperforms most deep learning based models in this setting.

On the contrary, CIFAR10 represents a much more significant challenge, as testified by the low performance of most models, possibly due to the poor image resolution and visual clutter between classes. We observe that our proposal is the only outperforming a simple KDE baseline; however, this finding should be put into perspective by considering the nature of non-parametric estimators. Indeed, non-parametric models access the whole training set for the evaluation of each sample. Consequently, despite they benefit large sample sets in terms of density modeling, they lead into an unfeasible inference as the dataset grows in size.

The possible reasons behind the difference in performance w.r.t. DAE are twofold. Firstly, DAE can recognize novel samples solely based on the reconstruction error, hence relying on its memorization capabilities, whereas our proposal also considers the likelihood of their representations under the learned prior, thus exploiting surprisal as well. Secondly, by minimizing the differential entropy of the latent distribution, our proposal increases the discriminative capability of the reconstruction. Intuitively, this last statement can be motivated observing that novelty samples are forced to reside in a high probability region of the latent space, the latter bounded to solely capture unsurprising factors of variation arising from the training set. On the other hand, the gap w.r.t. VAE suggests that, for the task at hand, a more flexible autoregressive prior should be preferred over the isotropic multivariate Gaussian. On this last point, VAE seeks representations whose average surprisal converges to a fixed and expected value (the differential entropy of its prior), whereas our solution minimizes such quantity within its MLE objective.

	UCSD Ped2	ShanghaiTech
MPPC+SFA [124]	0.613	-
ConvAE [56]	0.850	0.609
ConvLSTM-AE [121]	0.881	-
Hinami *et al.* [61]	0.922	-
TSC [122]	0.910	0.679
Stacked RNN [122]	0.922	0.680
FFP [111]	0.935	-
FFP+MC [111]	**0.954**	**0.728**
Ours	**0.954**	**0.725**

Table 3.3: AUROC performance of our model w.r.t. state-of-the-art competitors.

This flexibility allows modulating the richness of the latent representation vs. the reconstructing capability of the model. Differently, in VAEs, the fixed prior acts as a blind regularizer potentially leading to over-smooth representations.

Fig. 3.5 reports an ablation study questioning the loss functions aggregation presented in Eq. 3.9. The figure illustrates ROC curves under three different novelty scores: *i)* the log-likelihood term, *ii)* the reconstruction term, and *iii)* the proposed scheme that accounts for both. As highlighted in the picture, accounting for both memorization and surprisal aspects is advantageous in each dataset.

3.3.2 Video Novelty Detection

In video surveillance contexts, novelty is often considered in terms of abnormal human behavior. Thus, we evaluate our proposal against state-of-the-art anomaly detection models. For this purpose, we considered two standard benchmarks in literature, namely UCSD Ped2 [25] and ShanghaiTech [122]. Despite the differences in the number of videos and their resolution, they both contain anomalies that typically arise in surveillance scenarios (*e.g.*, vehicles in pedestrian walkways, pick-pocketing, brawling). For UCSD Ped, we preprocessed input clips of 16 frames to extract smaller patches and perturbed such inputs with random Gaussian noise with $\sigma = 0.025$. We compute the novelty score of each input clip as the mean novelty score among all patches. Concerning ShanghaiTech, we removed the dependency on the scenario by estimating the foreground for each frame of a clip with a standard MOG-based approach and removing the background. We

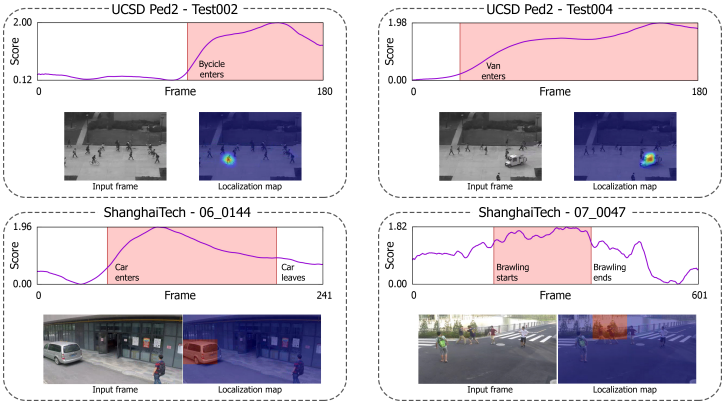


Figure 3.6: Novelty scores and localizations maps for samples drawn from UCSD Ped2 and ShanghaiTech. For each example, we report the trend of the assessed score, highlighting with a different color the time range in which an anomalous subject comes into the scene.

fed the model with 16-frames clips, but ground-truth anomalies are labeled at frame level. In order to recover the novelty score of each frame, we compute the mean score of all clips in which it appears. We then merge the two terms of the loss function following the same strategy illustrated in Eq. 3.9, computing however normalization coefficients in a per-sequence basis, following the standard approach in the anomaly detection literature. The scores for each sequence are then concatenated to compute the overall AUROC of the model. Additionally, we envision localization strategies for both datasets. To this aim, for UCSD, we denote a patch exhibiting the highest novelty score in a frame as anomalous. Differently, in ShanghaiTech, we adopt a sliding-window approach [212]: as expected, when occluding the source of the anomaly with a rectangular patch, the novelty score drops significantly.

Tab. 3.3 reports results in comparison with prior works. Despite a more general formulation, our proposal scores on-par with the current state-of-the-art solutions specifically designed for video applications and taking advantage of optical flow estimation and motion constraints. Indeed, in the absence of such hypotheses (FFP entry in Tab. 3.3), our method outperforms future frame prediction on UCSD Ped2. Finally, we refer the reader to Fig. 3.6 for several qualitative assessments regarding the novelty score and localization capabilities

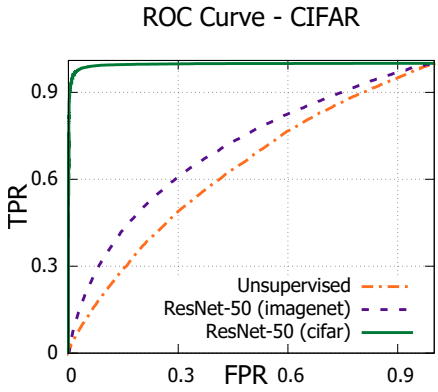


Figure 3.7: CIFAR-10 ROC curves with semantic input vectors. Each curve is an interpolation among the ten classes.

3.4 Model Analysis

CIFAR-10 with semantic features We investigate the behavior of our model in the presence of different assumptions regarding the expected nature of novel samples. We expect that, as the correctness of such assumptions increases, novelty detection performance will scale accordingly. Such a trait is particularly desirable for applications in which prior beliefs about novel examples can be envisioned. To this end, we leverage the CIFAR-10 benchmark described in Sec. 3.3.1 and change the type of information provided as input. Specifically, instead of raw images, we feed our model with semantic representations extracted by ResNet-50 [58], either pre-trained on Imagenet (*i.e.*, assume semantic novelty) or CIFAR-10 itself (*i.e.*, assume data-specific novelty). The two models achieved respectively 79.26 and 95.4 top-1 classification accuracies on the respective test sets. Even though this procedure is to be considered unfair in novelty detection, it serves as a sanity check delivering the upper-bound performance our model can achieve when applied to even better features. To deal with dense inputs, we employ a fully connected autoencoder and MFC layers within the estimation network.

Fig. 3.7 illustrates the resulting ROC curves, where semantic descriptors improve AUROC w.r.t. raw image inputs (entry "Unsupervised"). Such results suggest that our model profitably takes advantage of the separation between normal and abnormal input representations and scales accordingly, even up to optimal

CIFAR-10	
$\text{LSTM}_{[100]}$	0.623
$\text{LSTM}_{[32,32,32,32,100]}$	0.622
$\text{MFC}_{[100]}$	0.625
$\text{MFC}_{[32,32,32,32,100]}$	**0.641**

(a)

UCSD Ped2	
$\text{LSTM}_{[100]}$	0.849
$\text{LSTM}_{[4,4,4,4,100]}$	0.845
$\text{MSC}_{[100]}$	0.849
$\text{MSC}_{[4,4,4,4,100]}$	**0.954**

(b)

Figure 3.8: Comparison of different architectures for the autoregressive density estimation in feature space. We indicate with $\text{LSTM}_{[F_1,F_2,\ldots,F_N]}$ - same goes for MFC and MSC - the output shape for each of the N layers composing the estimator. Results are reported in terms of test AUROC.

performance for the task under consideration. Nevertheless, it is interesting to note how different degrees of supervision deliver significantly different results. As expected, dataset-specific supervision increases the AUROC from 0.64 up to 0.99 (a perfect score). Surprisingly, semantic feature vectors trained on Imagenet (which contains all CIFAR classes) provide a much lower boost, yielding an AUROC of 0.72. This indicates that, even when the semantic of novelty can be known in advance, its contribution has a limited impact in modeling the normality, mostly because novelty can depend on other cues (*e.g.*, low-level statistics).

Autoregression via recurrent layers To measure the contribution of the proposed MFC and MSC layers described in Sec. 3.2, we test on CIFAR-10 and UCSD Ped2, alternative solutions for the autoregressive density estimator. Specifically, we investigate recurrent networks, as they represent the most natural alternative featuring autoregressive properties. We benchmark the proposed building blocks against an estimator composed of LSTM layers, which is designed to sequentially observe latent symbols $\mathbf{z}_{<i}$ and output the CPD of z_i as the hidden state of the last layer. We test MFC, MSC and LSTM in single-layer and multi-layer settings, and report all outcomes in Fig. 3.8. Even though our solutions perform similarly to the recurrent baseline when employed in a shallow setting, they significantly take advantage of their depth when stacked in consecutive layers. MFC and MSC, indeed, employ disentangled parametrizations for each output CPD. This property is equivalent to the adoption of a specialized estimator network for each z_i, thus increasing the proficiency in modeling the density of its designated CPD. On the contrary, LSTM networks embed all the history (*i.e.*, the observed symbols) in

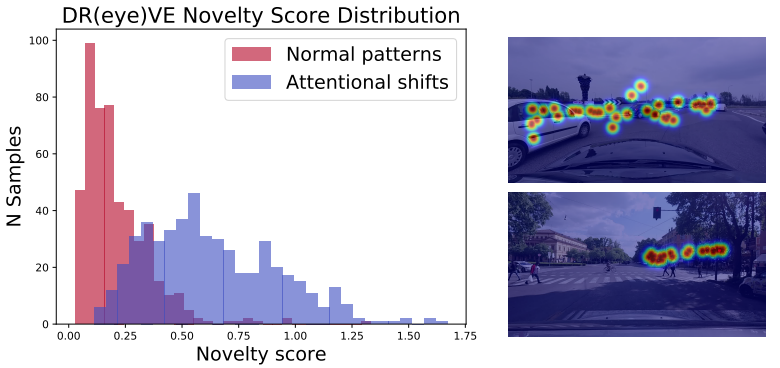


Figure 3.9: Left, the distribution of novelty scores assigned to normal patterns against attentional shifts labeled within the DR(eye)VE dataset. Right, DR(eye)VE clips yielding the highest novelty score (*i.e.*, clips in which the attentional pattern shifts from the expected behavior). Interestingly, they depict some peculiar situations such as waiting for the traffic light or approaching a roundabout.

their memory cells, but manipulate each input of the sequence through the same weight matrices. In this regime, the recurrent module needs to learn parameters shared among symbols, losing specialization and eroding its capabilities.

Novelty in cognitive temporal processes As a potential application of our proposal, we investigate its capability in modeling human attentional behavior. To this end, we employ the DR(eye)VE dataset [139], introduced for the prediction of focus of attention in driving contexts. It features 74 driving videos where frame-wise fixation maps are provided, highlighting the region of the scene attended by the driver. In order to capture the dynamics of attentional patterns, we purposely discard the visual content of the scene and optimize our model on clips of fixation maps, randomly extracted from the training set. After training, we rely on the novelty score of each clip as a proxy for the uncommonness of an attentional pattern. Moreover, since the dataset features annotations of peculiar and unfrequent patterns (such as distractions, recording errors), we can measure the correlation of the captured novelty w.r.t. those. In terms of AUROC, our model scores 0.926, highlighting that novelty can arise from unexpected behaviors of the driver, such as distractions or other shifts in attention. Fig. 3.9 reports the different distribution of novelty scores for ordinary and peculiar events.

3.5 Conclusion

In this chapter we have discussed a comprehensive framework for novelty detection, in which prior knowledge is embodied by an auxiliary component acting in latent space. Specifically, we formalize our model to capture the twofold nature of novelties, which concerns the incapability to remember unseen data and the surprisal aroused by the observation of their latent representations. From a technical perspective, both terms are modeled by a deep generative autoencoder, paired with an additional autoregressive density estimator learning the distribution of latent vectors by maximum likelihood principles. To this aim, two different masked layers have been introduced to account for image and video data. We have shown that the introduction of such an auxiliary module leads to the minimization of the encoder's differential entropy, which proves to be a suitable regularizer for the task at hand. Experimental results have indicated that state-of-the-art performance in one-class and anomaly detection settings, fostering the flexibility of our framework for different tasks without making any data-related assumption.

Chapter 4

Continual Learning

This chapter showcase how the exploitation of prior knowledge is rewarding when the model is asked to learn several classification tasks one after the other, in a sequential manner. In this respect, we would expect the model to acquire new knowledge on-the-fly, incorporating new classes with the current one. However, if the learning focuses on the current set of examples solely, a sudden performance deterioration occurs on the old data, referred to as **catastrophic forgetting** [130]. As a trivial workaround, one could store all incoming examples and re-train from scratch when needed, but this is often impracticable in terms of required resources.

The research field of **Continual Learning (CL)** aims at relieving catastrophic forgetting while limiting computational costs and memory footprint [130]. This chapter focuses on **General Continual Learning (GCL)**, which addresses the peculiarities of real-world applications, where memory is bounded and tasks intertwine and overlap. On this latter point, [37] has recently introduced a series of guidelines that CL methods should realize to be applicable in practice: *i)* **no task boundaries**: do not rely on boundaries between tasks during training; *ii)* **no test time oracle**: do not require task identifiers at inference time; *iii)* **constant memory**: have a bounded memory footprint throughout the entire training phase.

In the following, we show that GCL can be favorably addressed mixing rehearsal with knowledge distillation; our simple baseline, **Dark Experience Replay (DER)**, matches the network's responses sampled throughout the optimization trajectory, thus promoting consistency with its past. Such a regularization strategy – which uses past examples to impose prior knowledge over network outputs – outperforms consolidated approaches and leverages limited resources. We show that

this holds on both standard benchmarks and a novel evaluation setting (MNIST-360); we further explore its generalization capabilities, showing its regularization being beneficial beyond mere performance.

4.1 Preliminaries

Rehearsal-based methods Several approaches tackle catastrophic forgetting by replaying a subset of the training data stored in a memory buffer. Early works [147, 154] proposed **Experience Replay (ER)**, that is interleaving old samples with current data in training batches. Several recent studies directly expand on this idea: **Meta-Experience Replay (MER)** [151] casts replay as a meta-learning problem to maximize transfer from past tasks while minimizing interference; **Gradient based Sample Selection (GSS)** [4] introduces a variation on ER to store optimally chosen examples in the memory buffer; **Hindsight Anchor Learning (HAL)** [28] complements replay with an additional objective to limit forgetting on pivotal learned data-points. On the other hand, **Gradient Episodic Memory (GEM)** [117] and its lightweight counterpart **Averaged-GEM (A-GEM)** [29] leverage old training data to build optimization constraints to be satisfied by the current update step. These works show improvements over ER when confining the learning to a small portion of the training set (*e.g.*, 1k examples per task). However, we believe that this setting rewards *sample efficiency* – *i.e.*, making good use of the few shown examples – which represents a potential confounding factor for assessing catastrophic forgetting. Indeed, Sec. 4.3 reveals that the above-mentioned approaches are not consistently superior to ER when lifting these restrictions, which motivates our research in this kind of methods.

Knowledge Distillation Several approaches exploit Knowledge Distillation [63] to mitigate forgetting by appointing a past version of the model as a teacher. **Learning Without Forgetting (LwF)** [102] computes a smoothed version of the current responses for the new examples at the beginning of each task, minimizing their drift during training. A combination of replay and distillation can be found in **iCaRL** [149], which employs a buffer as a training set for a *nearest-mean-of-exemplars* classifier while preventing the representation from deteriorating in later tasks via a self-distillation loss term.

Other Approaches Regularization-based methods extend the loss function with a term that prevents network weights from changing, as done by **Elastic Weight**

Methods	PNN [157]	PackNet [126]	HAT [170]	ER [147, 151]	MER [151]	GSS [4]	GEM [117]	A-GEM [29]	HAL [28]	iCaRL [149]	FDR [16]	LwF [102]	SI [213]	oEWC [85]	**DER (ours)**	**DER++ (ours)**
Constant memory	–	–	–	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
No task boundaries	–	–	–	✓	✓	✓	–	✓	–	–	–	–	–	–	✓	✓
No test time oracle	–	–	–	✓	✓	✓	✓	✓	✓	✓	✓	–	✓	✓	✓	✓

Table 4.1: Continual learning approaches and their compatibility with the General Continual Learning major requirements [37].

Consolidation (EWC) [85], **online EWC (oEWC)** [168], **Synaptic Intelligence (SI)** [213] and **Riemmanian Walk (RW)** [27]. Architectural methods, on the other hand, devote distinguished sets of parameters to distinct tasks. Among these, **Progressive Neural Networks (PNN)** [157] instantiates new networks incrementally as novel tasks occur, resulting in a linearly growing memory requirement. To mitigate this issue, **PackNet** [126] and **Hard Attention to the Task (HAT)** [170] share the same architecture for subsequent tasks, employing a heuristic strategy to prevent intransigence by allocating additional units when needed.

Overall It is not always easy to have a clear picture of the merits of these works: due to subtle differences in the way methods are evaluated, many state-of-the-art approaches only stand out in the setting where they were originally conceived. Several recent papers [37, 45, 64, 190] address this issue and conduct a critical review of existing evaluation settings, leading to the formalization of three main experimental settings [64, 190]. By conducting an extensive comparison on them, we surprisingly observe that a simple Experience Replay baseline consistently outperforms cutting-edge methods in the considered settings. As reported in Tab. 4.1, ER also stands out being one of the few methods that are fully compliant with GCL. MER [151] and GSS [4] fulfill the requirements as well, but they suffer from a very long running time which hinders their applicability.

4.2 Proposed Approach

Formally, a CL classification problem is split in T tasks; during each task $t \in \{1, ..., T\}$ input samples x and their corresponding ground truth labels y are drawn from an i.i.d. distribution D_t. A function f, with parameters θ, is optimized on one task at a time in a sequential manner. We indicate the output logits

with $h_\theta(x)$ and the corresponding probability distribution over the classes with $f_\theta(x) \triangleq \mathrm{softmax}(h_\theta(x))$. The goal is to learn how to classify examples from any of the observed tasks up to the current one $t \in \{1, \dots, t_c\}$:

$$\underset{\theta}{\mathrm{argmin}} \sum_{t=1}^{t_c} \mathcal{L}_t, \quad \text{where} \quad \mathcal{L}_t \triangleq \mathbb{E}_{(x,y)\sim D_t}\big[\ell(y, f_\theta(x))\big]. \tag{4.1}$$

This is challenging as data from previous tasks are assumed to be unavailable: the best configuration of θ w.r.t. $\mathcal{L}_{1\dots t_c}$ must be sought without D_t for $t \in \{1, \dots, t_c - 1\}$. Ideally, we look for parameters that fit the current task well while approximating the behavior observed in the old ones: effectively, we encourage the network to mimic its original responses for past samples. To preserve the knowledge about previous tasks, we seek to minimize the following objective:

$$\mathcal{L}_{t_c} + \alpha \sum_{t=1}^{t_c-1} \mathbb{E}_{x\sim D_t}\big[\, D_{KL}(f_{\theta_t^*}(x) \;||\; f_\theta(x))\big], \tag{4.2}$$

where θ_t^* is the optimal set of parameters at the end of task t, and α is a hyper-parameter balancing the trade-off between the terms. This objective, which resembles the teacher-student approach, would require the availability of D_t for previous tasks. To overcome such a limitation, we introduce a replay buffer $\mathcal{M}_t$ holding past *experiences* for task t. Differently from other replay methods [4, 28, 151], we also retain the network's logits $z \triangleq h_{\theta_t}(x)$.

$$\mathcal{L}_{t_c} + \alpha \sum_{t=1}^{t_c-1} \mathbb{E}_{(x,z)\sim \mathcal{M}_t}\big[\, D_{KL}(\mathrm{softmax}(z) \;||\; f_\theta(x))\big]. \tag{4.3}$$

As we focus on General Continual Learning, we intentionally avoid relying on task boundaries to populate the buffer as the training progresses. Therefore, in place of the common task-stratified sampling strategy, we adopt *reservoir* sampling [198]: this way, we select $|\mathcal{M}|$ random samples from the input stream, guaranteeing that they have the same probability ${}^{|\mathcal{M}|}/_{|\mathcal{S}|}$ of being stored in the buffer, without knowing the length of the stream $\mathcal{S}$ in advance. We can rewrite Eq. 4.3 as follows:

$$\mathcal{L}_{t_c} + \alpha\, \mathbb{E}_{(x,z)\sim \mathcal{M}}\big[\, D_{KL}(\mathrm{softmax}(z) \;||\; f_\theta(x))\big]. \tag{4.4}$$

Such a strategy implies picking logits z during the optimization trajectory, so potentially different from the ones that can be observed at the task's local optimum.

Even if counter-intuitive, we empirically observed that this strategy does not hurt performance, while still being suitable without task boundaries. Furthermore, we observe that the replay of sub-optimal logits has beneficial effects in terms of flatness of the attained minima and calibration (see Sec. 4.4).

Under mild assumptions [63], the optimization of the KL divergence in Eq. 4.4 is equivalent to minimizing the Euclidean distance between the corresponding pre-softmax responses (*i.e.* logits). We opt for matching logits, as it avoids the information loss occurring in probability space due to the squashing function (*e.g.*, softmax) [115]. With these considerations in hands, Dark Experience Replay (DER, algorithm 1) optimizes the following objective:

$$\mathcal{L}_{t_c} + \alpha \, \mathbb{E}_{(x,z)\sim\mathcal{M}} \big[\|z - h_\theta(x)\|_2^2 \big]. \tag{4.5}$$

We approximate the expectation by computing gradients on buffer datapoints.

Dark Experience Replay++ It is worth noting that the *reservoir* strategy may weaken DER under some specific circumstances. Namely, when a sudden distribution shift occurs in the input stream, logits that are highly biased by the training on previous tasks might be sampled for later replay: leveraging the ground truth labels as well – as done by ER – could mitigate such a shortcoming. On these grounds, we also propose **Dark Experience Replay++ (DER++, algorithm 2)**, which equips the objective of Eq. 4.5 with an additional term on buffer datapoints, promoting higher conditional likelihood w.r.t. their ground truth labels:

$$\mathcal{L}_{t_c} + \alpha \, \mathbb{E}_{(x',y',z')\sim\mathcal{M}} \big[\|z' - h_\theta(x')\|_2^2 \big] \; + \; \beta \, \mathbb{E}_{(x'',y'',z'')\sim\mathcal{M}} \big[\ell(y'', f_\theta(x'')) \big], \tag{4.6}$$

where β is an additional coefficient balancing the last term. The model is not overly sensitive to α and β: setting them both to 0.5 yields stable performance.

4.2.1 Relation With Previous Works

While both DER and LWF [102] leverage knowledge distillation in Continual Learning, they adopt remarkably different approaches. The latter does not replay past examples, so it only encourages the similarity between teacher and student responses w.r.t. to datapoints of the current task. Alternatively, iCaRL [149] distills knowledge for past outputs w.r.t. past exemplars, which is more akin to our proposal. However, the former exploits the network appointed at the end of each task as the sole teaching signal. On the contrary, our methods store

Algorithm 1: Dark Experience Replay

Input: dataset D, parameters θ, scalar α, learning rate λ, $\mathcal{M} = \{\}$
for (x, y) **in** D **do**
  $(x', z', y') \leftarrow \textit{sample}(\mathcal{M})$
  $x_t,\ x'_t \leftarrow \textit{augment}(x),\ \textit{augment}(x')$
  $z \leftarrow h_\theta(x_t)$
  $reg \leftarrow \alpha \left\| z' - h_\theta(x'_t) \right\|_2^2$
  $\theta \leftarrow \theta + \lambda \cdot \nabla_\theta [\ell(y, f_\theta(x_t)) + reg]$
  $\mathcal{M} \leftarrow \textit{reservoir}(\mathcal{M}, (x, z))$
end for

Algorithm 2: Dark Experience Replay ++

Input: dataset D, parameters θ, scalars α and β, learning rate λ, $\mathcal{M} = \{\}$
for (x, y) **in** D **do**
  $(x', z', y') \leftarrow \textit{sample}(\mathcal{M})$
  $(x'', z'', y'') \leftarrow \textit{sample}(\mathcal{M})$
  $x_t \leftarrow \textit{augment}(x)$
  $x'_t,\ x''_t \leftarrow \textit{augment}(x'),\ \textit{augment}(x'')$
  $z \leftarrow h_\theta(x_t)$
  $reg \leftarrow \alpha \left\| z' - h_\theta(x'_t) \right\|_2^2 + \beta\, \ell(y'', f_\theta(x''_t))$
  $\theta \leftarrow \theta + \lambda \cdot \nabla_\theta [\ell(y, f_\theta(x_t)) + reg]$
  $\mathcal{M} \leftarrow \textit{reservoir}(\mathcal{M}, (x, z, y))$
end for

logits sampled throughout the optimization trajectory, resembling several different teacher parametrizations.

A close proposal to ours is given by **Function Distance Regularization (FDR)** for combatting catastrophic forgetting (Sec. 3.1 of [16]). Like FDR, we use past exemplars and network outputs to align past and current outputs. However, similarly to the iCaRL discussion above, FDR stores network responses at task boundaries and thus cannot be employed in a GCL setting. Instead, the experimental analysis we present in Sec. 4.4 reveals that the need of task boundaries can be relaxed through *reservoir* without experiencing a drop in performance; on the contrary we empirically observe that DER and DER++ achieve significantly superior results and remarkable properties. We finally highlight that the motivation behind [16] lies chiefly in studying how the training trajectory of NNs can be characterized in a functional L^2 Hilbert space, whereas the potential of function-space regularization

for Continual Learning problems is only coarsely addressed with a single experiment on MNIST. In this respect, we present extensive experiments on multiple CL settings as well as a detailed analysis (Sec. 4.4) providing a deeper understanding on the effectiveness of this kind of regularization.

4.3 Experiments

4.3.1 Datasets

We adhere to [64, 190] and model the sequence of tasks using three settings:

Task Incremental Learning (Task-IL) and **Class Incremental Learning (Class-IL)** split the training samples into partitions of classes (tasks). Although similar, the former provides task identities to select the relevant classifier for each example, whereas the latter does not; this difference makes Task-IL and Class-IL the easiest and hardest scenarios among the three [190]. In practice, we follow [37, 213] by splitting CIFAR-10 [88] and Tiny ImageNet [175] in 5 and 10 tasks, each of which introduces 2 and 20 classes respectively. We show all the classes in the same fixed order across different runs.

Domain Incremental Learning (Domain-IL) feeds all classes to the network during each task, but applies a task-dependent transformation to the input; task identities remain unknown at test time. For this setting, we leverage two common protocols built upon the MNIST dataset [96], namely **Permuted MNIST** [85] and **Rotated MNIST** [117]. They both require the learner to classify all MNIST digits for 20 subsequent tasks, but the former applies a random permutation to the pixels, whereas the latter rotates the images by a random angle in the interval $[0, \pi)$.

As done in previous works [45, 149, 190, 204], we provide task boundaries to the competitors demanding them at training time (*e.g.* oEWC or LwF). This choice is meant to ensure a fair comparison between our proposal – which does not need boundaries – and a broader class of methods in literature.

4.3.2 Evaluation Protocol

Architecture For tests we conducted on variants of the MNIST dataset, we follow [117, 151] by employing a fully-connected network with two hidden layers, each one comprising of 100 ReLU units. For CIFAR-10 and Tiny ImageNet, we follow [149] and rely on ResNet18 [58] (not pre-trained).

Augmentation For CIFAR-10 and Tiny ImageNet, we apply random crops and horizontal flips to both stream and buffer examples. We propagate this choice to competitors for fairness. It is worth noting that combining data augmentation with our regularization objective enforces an implicit consistency loss [6, 15], which aligns predictions for the same example subjected to small data transformations.

Hyperparameter selection We select hyperparameters by performing a grid-search on a validation set, the latter obtained by sampling 10% of the training set. For the Domain-IL scenario, we make use of the final average accuracy as the selection criterion. Differently, we perform a combined grid-search for Class-IL and Task-IL, choosing the configuration that achieves the highest final accuracy averaged on the two settings.

Training To provide a fair comparison among CL methods, we train all the networks using the Stochastic Gradient Descent (SGD) optimizer. Despite being interested in an online scenario, with no additional passages on the data, we reckon it is necessary to set the number of epochs per task in relation to the dataset complexity. Indeed, if even the pure-SGD baseline fails at fitting a single task with adequate accuracy, we could not properly disentangle the effects of catastrophic forgetting from those linked to underfitting[1]. For MNIST-based settings, one epoch per task is sufficient. Conversely, we increase the number of epochs to 50 for Sequential CIFAR-10 and 100 for Sequential Tiny ImageNet respectively, as commonly done by works that test on harder datasets [149, 204, 213]. We deliberately hold batch size and minibatch size out from the hyperparameter space, thus avoiding the flaw of a variable number of update steps for different methods.

4.3.3 Experimental Results

In this section, we compare DER and DER++ against two regularization-based methods (*oEWC*, *SI*), two methods leveraging Knowledge Distillation (*iCaRL*, *LwF*[2]), one architectural method (*PNN*) and six rehearsal-based methods (*ER*, *GEM*, *A-GEM*, *GSS*, *FDR* [16], *HAL*)[3]. We further provide a lower bound, consisting of *SGD* without any countermeasure to forgetting and an upper bound given

[1] We refer the reader to Sec. 4.3.3 for an experimental discussion regarding this issue

[2] In Class-IL, we adopted a multi-class implementation as done in [149].

[3] We omit MER as we experienced an intractable training time on these benchmarks (*e.g.* while DER takes approximately 2.5 hours on Seq. CIFAR-10, MER takes 300 hours – see Sec. 4.4 for further comparisons).

Buffer	Method	S-CIFAR-10		S-Tiny-ImageNet		P-MNIST	R-MNIST
		Class-IL	*Task-IL*	*Class-IL*	*Task-IL*	*Domain-IL*	*Domain-IL*
–	JOINT	92.20±0.15	98.31±0.12	59.99±0.19	82.04±0.10	94.33±0.17	95.76±0.04
	SGD	19.62±0.05	61.02±3.33	7.92±0.26	18.31±0.68	40.70±2.33	67.66±8.53
–	oEWC [168]	**19.49±0.12**	68.29±3.92	7.58±0.10	19.20±0.31	**75.79±2.25**	**77.35±5.77**
	SI [213]	19.48±0.17	68.05±5.91	6.58±0.31	36.32±0.13	65.86±1.57	71.91±5.83
	LwF [102]	19.46±0.31	63.65±1.80	**8.57±0.11**	16.57±0.37	-	-
	PNN [157]	-	**95.13±0.72**	-	**67.84±0.29**	-	-
200	ER [151]	48.33±1.57	91.49±0.92	8.77±0.17	38.97±0.97	72.37±0.87	85.01±1.90
	GEM [117]	25.54±0.76	90.44±0.94	-	-	66.93±1.25	80.80±1.15
	A-GEM [29]	20.04±0.34	83.88±1.49	8.07±0.08	22.77±0.03	66.42±4.00	81.91±0.76
	iCaRL [149]	60.58±1.32	**93.97±0.53**	**14.72±0.59**	**42.84±0.92**	-	-
	FDR [16]	30.91±2.74	91.01±0.68	8.70±0.19	40.36±0.68	74.77±0.83	85.22±3.35
	GSS [4]	39.07±5.59	88.80±2.89	-	-	63.72±0.70	79.50±0.41
	HAL [28]	34.90±2.55	83.14±3.66	-	-	74.15±1.65	84.02±0.98
	DER (ours)	61.93±1.79	91.40±0.92	11.87±0.78	40.22±0.67	81.74±1.07	90.04±2.61
	DER++ (ours)	**64.88±1.17**	91.92±0.60	10.96±1.17	40.87±1.16	**83.58±0.59**	**90.43±1.87**
500	ER [151]	60.98±1.48	**94.19±0.32**	11.06±0.32	49.89±0.73	80.60±0.86	88.91±1.44
	GEM [117]	26.20±1.26	92.16±0.69	-	-	76.88±0.52	81.15±1.98
	A-GEM [29]	22.67±0.57	89.48±1.45	8.06±0.04	25.33±0.49	67.56±1.28	80.31±6.29
	iCaRL [149]	55.42±4.16	91.43±1.84	**20.18±0.56**	**52.07±0.58**	-	-
	FDR [16]	28.71±3.23	93.29±0.59	10.54±0.21	49.88±0.71	83.18±0.53	89.67±1.63
	GSS [4]	49.73±4.78	91.02±1.57	-	-	76.00±0.87	81.58±0.58
	HAL [28]	46.19±4.14	86.08±2.48	-	-	80.13±0.49	85.00±0.96
	DER (ours)	70.51±1.67	93.40±0.39	17.75±1.14	51.78±0.88	87.29±0.46	92.24±1.12
	DER++ (ours)	**72.70±1.36**	93.88±0.50	19.38±1.41	51.91±0.68	**88.21±0.39**	**92.77±1.05**
5120	ER [151]	84.30±0.73	**97.02±0.15**	29.93±0.47	67.89±0.50	89.90±0.13	93.45±0.56
	GEM [117]	25.26±3.46	95.55±0.02	-	-	87.42±0.95	88.57±0.40
	A-GEM [29]	21.99±2.29	90.10±2.09	7.96±0.13	26.22±0.65	73.32±1.12	80.18±5.52
	iCaRL [149]	63.47±1.33	95.47±0.26	31.60±0.33	64.54±0.30	-	-
	FDR [16]	19.70±0.07	94.32±0.97	28.97±0.41	68.01±0.42	90.87±0.16	94.19±0.44
	GSS [4]	67.27±4.27	94.19±1.15	-	-	82.22±1.14	85.24±0.59
	HAL [28]	64.99±3.71	89.01±2.64	-	-	89.20±0.14	91.17±0.31
	DER (ours)	83.81±0.33	95.43±0.33	36.73±0.64	69.50±0.26	91.66±0.11	94.14±0.31
	DER++ (ours)	**85.24±0.49**	96.12±0.21	**39.02±0.97**	**69.84±0.63**	**92.26±0.17**	**94.65±0.33**

Table 4.2: Classification results for standard CL benchmarks, averaged across 10 runs. '-' indicates experiments we were unable to run, because of compatibility issues (*e.g.* between PNN, iCaRL and LwF in Domain-IL) or intractable training time (*e.g.* GEM, HAL or GSS on Tiny ImageNet).

by training all tasks jointly (JOINT). Tab. 4.2 reports performance in terms of average accuracy at the end of all tasks; results are averaged across ten runs, each one involving a different initialization.

DER and DER++ achieve state-of-the-art performance in almost all settings. When compared to oEWC and SI, the gap appears unbridgeable, suggesting that regularization towards old sets of parameters does not suffice. We argue that this is due to local information modeling weights importance: as it is computed in earlier tasks, it could become untrustworthy in later ones. While being computationally more efficient, LWF performs worse than SI and oEWC on average. PNN, which achieves the strongest results among non-rehearsal methods, attains lower accuracy than replay-based ones despite its memory footprint being much higher.

When compared to rehearsal methods, DER and DER++ show strong performance in the majority of benchmarks, especially in the Domain-IL scenario. For these problems, a shift occurs within the input domain, but not within the classes: hence, the relations among them also likely persist. As an example, if it is true that during the first task number 2's visually look like 3's, this still holds when applying rotations or permutations, as it is done in the following tasks. We argue that leveraging soft-targets in place of hard ones (ER) carries more valuable information [63], exploited by DER and DER++ to preserve the similarity structure through the data-stream. Additionally, we observe that methods resorting to gradients (GEM, A-GEM, GSS) seem to be less effective in this setting.

The gap we observe in Domain-IL is also found in the Class-IL setting, as DER is remarkably capable of learning how classes from different tasks are related to each other. This is not so relevant in Task-IL, where DER performs on par with ER on average. In it, classes only need to be compared in exclusive subsets, and maintaining an overall vision is not especially rewarding. In such a scenario, DER++ manages to effectively combine the strengths of both methods, resulting in generally better accuracy. Interestingly, iCaRL appears valid when using small buffers; we believe this is due to its helpful *herding* strategy, ensuring that all classes are equally represented in memory. As a side note, other ER-based methods (HAL and GSS) show weaker results than ER itself on such challenging datasets.

Single-Epoch Setting Several Continual Learning works present experiments even on fairly complex datasets (*e.g.:* CIFAR-10, CIFAR-100, Mini ImageNet) in which the model is only trained for one epoch for each task [4, 28, 29, 117]. As showing the model each example only once could be deemed closer to real-world CL scenarios, this is a very compelling setting and somewhat close in spirit to the reasons why we focus on General Continual Learning.

	Buffer	ER	FDR	DER++	JOINT	JOINT
#epochs		1	1	1	1	50/100
Seq. CIFAR-10	200	37.64	21.22	**41.93**	56.74	92.20
	500	45.22	21.06	**48.04**		
	5120	50.28	20.57	**53.31**		
Seq. Tiny ImageNet	200	5.98	4.87	**6.35**	19.37	59.99
	500	8.39	4.76	**8.65**		
	5120	16.04	4.96	**16.41**		

Table 4.3: Single-epoch evaluation setting (Class-IL).

However, we see that committing to just one epoch (hence, few gradient steps) makes it difficult to disentangle the effects of catastrophic forgetting from those of underfitting. This is especially relevant when dealing with complex datasets and deserves further investigation: for this reason, we conduct a single-epoch experiment on Seq. CIFAR-10 and Seq. Tiny ImageNet. We include in Tab. 4.3 the performance of different rehearsal methods; additionally, we report the results of joint training when limiting the number of epochs to one and, *vice versa*, when such limitation is removed (see last two columns). While the multi-epoch joint training achieves a satisfactory accuracy, the single-epoch counterpart (which is the upper bound of the other CL methods) yields a much lower accuracy and underfits dramatically. In light of this, it is hard to evaluate the merits of CL methods, whose evaluation is severely undermined by this confounding factor. Although DER++ proves reliable even in this setting, we feel that future works should strive for realism by designing experimental settings which are fully in line with the guidelines of GCL [37] rather than adopting the single-epoch protocol.

MNIST-360 To address the General Continual Learning desiderata, we propose a novel protocol: MNIST-360. It models a stream of data presenting batches of two consecutive MNIST digits at a time (*e.g.* $\{0, 1\}$, $\{1, 2\}$, $\{2, 3\}$ etc.), as depicted in Fig. 4.1. We rotate each example of the stream by an increasing angle and, after a fixed number of steps, switch the lesser of the two digits with the following one. As it is impossible to distinguish 6's and 9's upon rotation, we do not use 9's in MNIST-360. The stream visits the nine possible couples of classes three times, allowing the model to leverage positive transfer when revisiting a previous task. Moreover, we guarantee that: *i)* each example is shown once during training; *ii)*

Figure 4.1: Example batches of the MNIST-360 stream.

JOINT	SGD	Buffer	ER [151]	MER [151]	A-GEM-R [29]	GSS [4]	DER	DER++
$82.98_{\pm 3.24}$	$19.09_{\pm 0.69}$	200	$49.27_{\pm 2.25}$	$48.58_{\pm 1.07}$	$28.34_{\pm 2.24}$	$43.92_{\pm 2.43}$	$\mathbf{55.22_{\pm 1.67}}$	$54.16_{\pm 3.02}$
		500	$65.04_{\pm 1.53}$	$62.21_{\pm 1.36}$	$28.13_{\pm 2.62}$	$54.45_{\pm 3.14}$	$69.11_{\pm 1.66}$	$\mathbf{69.62_{\pm 1.59}}$
		1000	$75.18_{\pm 1.50}$	$70.91_{\pm 0.76}$	$29.21_{\pm 2.62}$	$63.84_{\pm 2.09}$	$75.97_{\pm 2.08}$	$\mathbf{76.03_{\pm 1.61}}$

Table 4.4: Accuracy on the test set for MNIST-360.

two digits of the same class are never observed under the same rotation.

It is noted that such a setting presents both sharp (change in class) and smooth (rotation) distribution shifts; therefore, for the algorithms relying on explicit boundaries, it would be hard to identify them. As outlined in Sec. 4.1, just ER, MER, and GSS are suitable for GCL. However, we also explore a variant of A-GEM equipped with a reservoir memory buffer (A-GEM-R). We compare these approaches with DER and DER++, reporting the results in Table 4.4 (we keep the same fully-connected network used on MNIST-based datasets). As can be seen, DER and DER++ outstand in such a challenging scenario, supporting the effectiveness of the proposed baselines against alternative replay methods.

4.4 Model Analysis

We provide here an in depth analysis of DER and DER++ by comparing them against FDR and ER. By so doing, we gather insights on logits sampled throughout the optimization trajectory, as opposed to ones at task boundaries and true labels.

DER converges to flatter minima Recent studies [26, 72, 78] link Deep Network generalization to the geometry of the loss function, namely the flatness of the attained minimum. While these works link flat minima to good train-test generalization, here we are interested in examining their weight in Continual Learning. Let us suppose that the optimization converges to a sharp minimum w.r.t. $\mathcal{L}_{1\dots t_c}$ (Eq. 4.1): in that case, the tolerance towards local perturbations is quite low. As a side effect, the drift we will observe in parameter space (due to the optimization of $\mathcal{L}_{1\dots t'}$ for $t' > t_c$) will intuitively lead to an even more serious drop in performance.

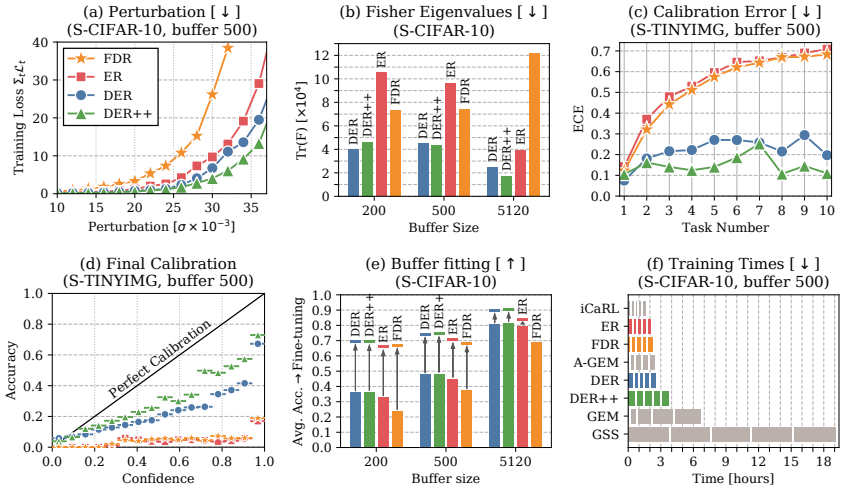


Figure 4.2: Results for the model analysis. [↑] higher is better, [↓] lower is better.

On the contrary, reaching a flat minimum for $\mathcal{L}_{1\ldots t_c}$ could give more room for exploring neighbouring regions of the parameter space, where one may find a new optimum for task t' without experiencing a severe failure on tasks $1, \ldots, t_c$. We conjecture that the effectiveness of the proposed baseline is linked to its ability to attain flatter and robust minima, which generalizes better to unseen data and, additionally, favors adaptability to incoming tasks. To validate this hypothesis, we compare the flatness of the training minima of FDR, ER, DER and DER++ utilizing two distinct metrics.

Firstly, as done in [215, 219], we consider the model at the end of training and add independent Gaussian noise with growing σ to each parameter. This allows us to evaluate its effect on the average loss across all training examples. As shown in Fig. 4.2(a) (S-CIFAR-10, buffer size 500), ER and especially FDR reveal higher sensitivity to perturbations than DER and DER++. Furthermore, [26, 72, 78] propose measuring flatness by evaluating the eigenvalues of $\nabla^2_\theta \mathcal{L}$: sharper minima correspond to larger Hessian eigenvalues. At the end of training on S-CIFAR-10, we compute the empirical Fisher Information Matrix $F = \sum \nabla_\theta \mathcal{L} \, \nabla_\theta \mathcal{L}^T / N$ w.r.t. the whole training set (as an approximation of the intractable Hessian [26, 85]). Fig. 4.2(b) reports the sum of its eigenvalues $\mathrm{Tr}(F)$: as one can see, DER and especially DER++ produce the lowest eigenvalues, which translates into flatter

minima following our intuitions. It is worth noting that FDR's large $\mathrm{Tr}(F)$ for buffer size 5120 could be linked to its failure case in S-CIFAR-10, Class-IL.

DER converges to more calibrated networks Calibration is a desirable property for a learner, measuring how much the confidence of its predictions corresponds to its accuracy. Ideally, we expect output distributions whose shapes mirror the probability of being correct, thus quantifying how much one can trust a specific prediction. Recent works find out that modern Deep Networks – despite largely outperforming the ones from a decade ago – are less calibrated [54], as they tend to yield overconfident predictions [90]. In real-world applications, AI tools should support decisions in a continuous and online fashion (*e.g.* weather forecasting [21] or econometric analysis [52]); therefore, calibration represents an appealing property for any CL system aiming for employment outside of a laboratory environment.

Fig. 4.2(c, d) shows, for TinyImageNet, the value of the Expected Calibration Error (ECE) [133] during the training and the reliability diagram at the end of it respectively. It can be seen that DER and DER++ achieve a lower ECE than ER and FDR without further application of *a posteriori* calibration methods (*e.g.*, Temperature Scaling, Dirichlet Calibration, ...). This means that models trained using Dark Experience are less overconfident and, therefore, easier to interpret. As a final remark, *Liu et al.* link this property to the capability to generalize to novel classes in a zero-shot scenario [110], which could translate into an advantageous starting point for the subsequent tasks for DER and DER++.

On the informativeness of DER's buffer Network responses provide a rich description of the corresponding data point. Following this intuition, we posit that the merits of DER also result from the knowledge inherent in its memory buffer: when compared to the one built by ER, the former represents a more informative summary of the overall (full) CL problem. If that were the case, a new learner trained only on the buffer would yield an accuracy that is closer to the one given by jointly training on all data. To validate this idea, we train a network from scratch using the memory buffer as the training set: we can hence compare how memories produced by DER, ER, and FDR summarize well the underlying distribution. Fig. 4.2(e) shows the accuracy on the test set: as can be seen, DER delivers the highest performance, surpassing ER, and FDR. This is particularly evident for smaller buffer sizes, indicating that DER's buffer should be especially preferred in scenarios with severe memory constraints.

Further than its pure performance, we assess whether a model trained on the buffer can be specialized to an already seen task: this would be the case of new examples from an old distribution becoming available on the stream. We simulate it by sampling 10 samples per class from the test set and then fine-tuning on them with no regularization; Fig. 4.2 reports the accuracy on the remainder of the test set of each task: here too, DER's buffer yields better performance, thus providing additional insight regarding its representation capabilities.

On training time When facing up with a data-stream, one often cares about reducing the overall processing time: otherwise, training would not keep up with the rate at which data are made available to the stream. In this regard, we assess the performance of both DER and DER++ and other rehearsal methods in terms of wall-clock time (seconds) at the end of the last task. To guarantee a fair comparison, we conduct all tests under the same conditions, running each benchmark on a Desktop Computer equipped with an NVIDIA Titan X GPU and an Intel i7-6850K CPU. Fig. 4.2(f) reports the execution time we measured on S-CIFAR10, indicating the time necessary for each of 5 tasks. We draw the following remarks: *i)* DER has a comparable running time w.r.t. other replay methods such as ER, FDR, and A-GEM; *ii)* the time complexity for GEM grows linearly w.r.t. the number of previously seen tasks; *iii)* GSS is extremely slow (0.73 examples per second on average, while DER++ processes 3.71 examples per second), making it hardly viable in practical scenarios.

4.5 Conclusion

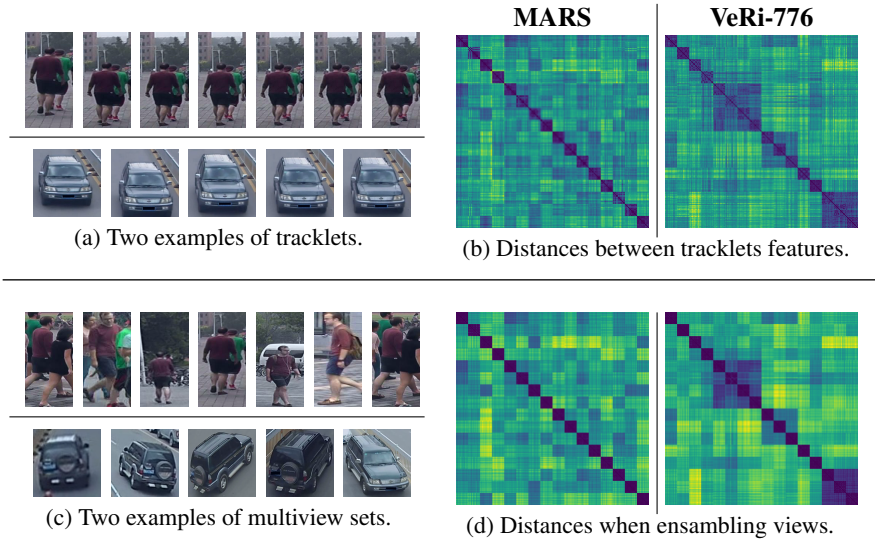
In this chapter, we have introduced Dark Experience Replay: a simple baseline for Continual Learning, which leverages Knowledge Distillation for retaining past experience and therefore avoiding catastrophic forgetting. We have shown the effectiveness of our proposal through an extensive experimental analysis, carried out on top of standard benchmarks. Also, we have argued that the recently formalized General Continual Learning provides the foundation for advances in diverse applications; for this reason, we have proposed MNIST-360 as an experimental protocol for this setting. We recommend DER as a starting point for future studies on both CL and GCL in light of its strong results on all evaluated settings and of the properties observed in Sec. 4.4.

Chapter 5

Person Re-Identification

Recent advances on Metric Learning [166, 174, 201, 189] give to researchers the foundation for computing suitable distance metrics between data points. In this context, Re-Identification (Re-ID) – which aims at associating images or videos of the same entity taken from different angles and cameras – has greatly benefited in diverse domains [222, 79, 164], as the common paradigm requires distance measures exhibiting robustness to variations in background clutters, as well as different viewpoints. To meet these criteria, various deep learning based approaches leverage videos to provide detailed descriptions for both query and gallery items. However, such a setting – known as Video-To-Video (V2V) Re-ID – does not represent a viable option in many scenarios (*e.g.* surveillance) [214, 205, 135, 53], where the query comprises a single image (Image-To-Video, I2V).

As observed in [53], a large gap in Re-ID performance still subsists between V2V and I2V, highlighting the number of query images as a critical factor in achieving good results. Contrarily, we advise the learnt representation should not be heavily affected when few images are shown to the network (*e.g.* only one). To bridge such a gap, [53, 18] propose a teacher-student paradigm, in which the student – in contrast with the teacher – has access to a small fraction of the frames in the video. Since the student is educated to mimic the output space of its teacher, it will show higher generalisation properties than its teacher when a single frame is available. It is noted that these approaches rely on transferring *temporal* information: as datasets often come with tracking annotation, they can guide the transfer from a tracklet into one of its frames. In this respect, we argue the limits of transferring temporal information: in fact, it is reasonable to assume high



(a) Two examples of tracklets.

(b) Distances between tracklets features.

(c) Two examples of multiview sets.

(d) Distances when ensambling views.

Figure 5.1: Visual comparison between tracklets and viewpoints variety, on person (MARS [221]) and vehicle (VeRi-776 [113]) re-id. Right: pairwise distances computed on top of features from ResNet-50. Inputs batches comprise 192 sets from 16 different identities, grouped by ground truth identity along each axis.

correlation between frames from the same tracklet (Fig. 5a), which may potentially underexploit the transfer. Moreover, limiting the analysis to the temporal domain does not guarantee robustness to variation in background appearances.

Here, we make a step forward and consider which information to transfer, shifting the paradigm from *time* to *views*: we argue that more valuable information arises when ensembling diverse views of the same target (Fig. 5c). This information often comes for free, as various datasets [221, 203, 113, 17] provide images capturing the same target from different camera viewpoints. To support our claim, Fig. 5 (right) reports pairwise distances computed on top of ResNet-50, when trained on Person and Vehicle Re-ID. In more details: matrices from Fig. 5b visualise the distances when tracklets are provided as input, whereas Fig. 5d shows the same for sets of views. As one can see, leveraging different views leads to a more distinctive blockwise pattern: namely, activations from the same identity are more consistent if compared to the ones computed in the tracklet scenario.

As shown in [186], this reflects a higher capacity to capture the semantics of the dataset, and therefore a *graceful* knowledge a teacher can transfer to a student.

Based on the above, we propose Views Knowledge Distillation (**VKD**), which transfers the knowledge lying in several views in a teacher-student fashion. VKD devises a two-stage procedure, which pins the visual variety as a teaching signal for a student who has to recover it using fewer views. We remark the following contributions: *i)* the student outperforms its teacher by a large margin, especially in the Image-To-Video setting; *ii)* a thorough investigation shows that the student focuses more on the target compared to its teacher and discards uninformative details; *iii)* importantly, we do not limit our analysis to a single domain, but instead achieve strong results on Person, Vehicle and Animal Re-ID.

5.1 Preliminaries

Image-To-Video Re-Identification The I2V Re-ID task has been successfully applied to multiple domains. In person Re-ID, [200] frames it as a point-to-set task, where image and video domains are aligned using a single deep network. The authors of [214] exploit time information by aggregating frames features via a Long-Short Term Memory. Eventually, a dedicated sub-network aggregates video features and match them against single image query ones. Authors of MGAT [11] employ a Graph Neural Network to model relationships between samples from different identities, thus enforcing similarity in the feature space. Dealing with vehicle Re-ID, authors from [114] introduce a large-scale dataset (VeRi-776) and propose PROVID and PROVID-BOT, which combine appearance and plate information in a progressive fashion. Differently, RAM [112] exploits multiple branches to extract global and local features, imposing a separate supervision on each branch and devising an additional one to predict vehicle attributes. VAMI [226] employs a viewpoint aware attention model to select core regions for different viewpoints. At inference time, they obtain a multiview descriptor through a conditional generative network, inferring information regarding the unobserved viewpoints. Differently, our approach asks the student to do it implicitly and in a lightweight fashion, thus avoiding the need for additional modules. Similarly to VAMI, [31] predicts the vehicle viewpoint along with appearance features; at inference, the framework provides distances according to the predicted viewpoint.

Knowledge Distillation Knowledge Distillation has been first investigated in [156, 63, 210] for model compression: the idea is to instruct a lightweight

model (student) to mimic the capabilities of a deeper one (teacher): as a gift, one could achieve both an acceleration in inference time as well as a reduction in memory consumption, without experiencing a large drop in performance. The approach discussed in this chapter uses the same techniques proposed in [63, 186] but for a different purpose: we are not primarily engaged in educating a light-weight module, but on improving the original model itself. In this framework – often called *self-distillation* [49, 206] – the transfer occurs from the teacher to a student with the same architecture, with the aim of improving the overall performance at the end of the training. Here, we get a step ahead and introduce an asymmetry between the teacher and student, which has access to fewer frames. In this respect, our approach closely relates to what [18] devises for Video Classification. Besides facing another task, a key difference subsists: while [18] limits the transfer along the temporal axis, our proposal advocates for distilling many views into fewer ones. On this latter point, we shall show that the teaching signal can be further enhanced when opening to diverse camera viewpoints. In the Re-Identification field, Temporal Knowledge Propagation (TKP) [53] similarly exploits intra-tracklet information to encourage the image-level representations to approach the video-level ones. In contrast with TKP: *i)* we do not rely on matching internal representations but instead their distances solely, thus making our proposal viable for cross-architecture transfer too; *ii)* at inference time, we make use of a single shared network to deal with both image and video domains, thus halving the number of parameters; *iii)* during transfer, we benefit from a larger visual variety, emerging from several viewpoints.

5.2 Proposed Approach

We pursue the aim of learning a function $\mathcal{F}_\theta(\mathcal{S})$ mapping a set of images $\mathcal{S} = (s_1, s_2, ..., s_n)$ into a representative embedding space. Specifically, $\mathcal{S}$ is a sequence of bounding boxes crops depicting a target (*e.g.* a person or a car), for which we are interested in inferring its corresponding identity. We take advantage of Convolutional Neural Networks (CNNs) for modelling $\mathcal{F}_\theta(\mathcal{S})$. Here, we look for two distinctive properties, aspiring to representations that are *i)* invariant to differences in background and viewpoint and *ii)* robust to a reduction in the number of query images. To achieve this, our proposal frames the training algorithm as a two-stage procedure, as follows:

- **First step** (Sec. 5.2.1): the backbone network is trained for the standard Video-To-Video setting.

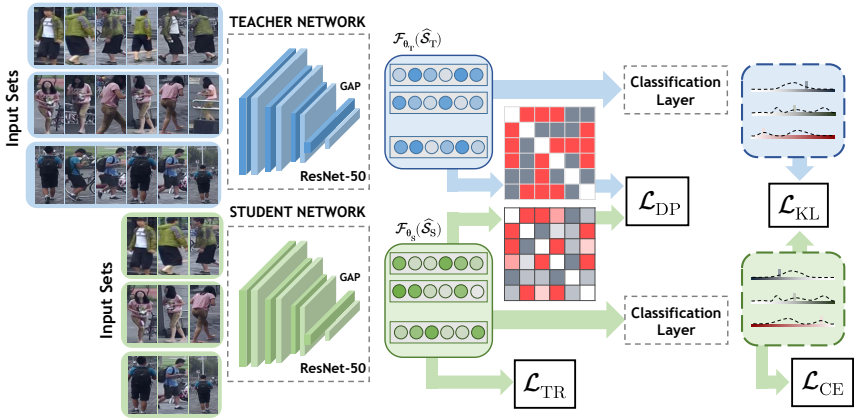


Figure 5.2: An overview of Views Knowledge Distillation (VKD): a student network is optimised to mimic the behaviour of its teacher using fewer views.

- **Second step** (Sec. 5.2.2): we appoint it as the teacher and freeze its parameters. Then, a new network with the role of the student is instantiated: we feed frames representing different views as input to the teacher and ask the student to mimic the same outputs from fewer frames (see Fig. 5.2).

5.2.1 Teacher Network

Without loss of generality, we will refer to ResNet-50 [58] as the backbone network, namely a module $f_\theta : \mathbb{R}^{\mathrm{W}\times\mathrm{H}\times 3} \mapsto \mathbb{R}^{\mathrm{D}}$ mapping each image s_i from S to a fixed-size representation d_i (in this case $D = 2048$). Following previous works [120, 53], we initialise the network weights on ImageNet and additionally include few amendments [120] to the architecture. First, we discard both the last ReLU activation function and final classification layer in favour of the BNNeck one [120] (*i.e.* batch normalisation followed by a linear layer). Second: to benefit from fine-grained spatial details, the stride of the last residual block is decreased from 2 to 1.

Set representation Given a set of images S, several solutions [116, 214, 107] may be assessed for designing the aggregation module, which fuses a variable-length set of representations $d_1, d_2, \ldots, d_n$ into a single one. Here, we naively

compute the set-level embedding $\mathcal{F}(\mathcal{S})$ through a temporal average pooling. While we acknowledge better aggregation modules exist, we do not place our focus on devising a new one, but instead on improving the earlier features extractor.

Teacher optimisation We train the base network - which will be the teacher during the following stage - combining a classification term $\mathcal{L}_{\text{CE}}$ (cross-entropy) with the triplet loss $\mathcal{L}_{\text{TR}}$[1]. The first can be formulated as:

$$\mathcal{L}_{\text{CE}} = -\boldsymbol{y} \log \hat{\boldsymbol{y}} \tag{5.1}$$

where $\boldsymbol{y}$ and $\hat{\boldsymbol{y}}$ represent the one-hot labels (identities) and the output of the softmax respectively. The second term $\mathcal{L}_{\text{TR}}$ encourages distance constraints in feature space, moving closer representations from the same target and pulling away ones from different targets. Formally:

$$\mathcal{L}_{\text{TR}} = \ln(1 + e^{\mathcal{D}\left(\mathcal{F}_\theta(\mathcal{S}_a^i), \mathcal{F}_\theta(\mathcal{S}_p^i)\right) - \mathcal{D}\left(\mathcal{F}_\theta(\mathcal{S}_a^i), \mathcal{F}_\theta(\mathcal{S}_n^j)\right)}), \tag{5.2}$$

where $\mathcal{S}_p$ and $\mathcal{S}_n$ are the hardest positive and negative for an anchor $\mathcal{S}_a$ within the batch. In doing so, we rely on the batch hard strategy [60] and include P identities coupled with K samples in each batch. Importantly, each set $\mathcal{S}^i$ comprises images drawn from the same tracklet [107, 48].

5.2.2 Views Knowledge Distillation (VKD)

After training the teacher, we propose to enrich its representation capabilities, especially when only few images are made available to the model. To achieve this, our proposal bets on the knowledge we can gather from different views, depicting the same object under different conditions. When facing re-identification tasks, one can often exploit camera viewpoints [221, 152, 113] to provide a larger variety of appearances for the target identity. Ideally, we would like to teach a new network to recover such a variety even from a single image. Since this information may not be inferred from a single frame, this can lead to an ill-posed task. Still, one can underpin this knowledge as a supervision signal, encouraging the student to focus on important details and favourably discover new ones. On this latter point, we refer the reader to Sec. 5.4 for a comprehensive discussion.

Views Knowledge Distillation (**VKD**) stresses this idea by forcing a student network $\mathcal{F}_{\theta_S}(\cdot)$ to match the outputs of the teacher $\mathcal{F}_{\theta_T}(\cdot)$. In doing so, we: *i)*

[1] For the sake of clarity, all the loss terms are referred to one single example. In the implementation, we extend the penalties to a batch by averaging.

allow the teacher to access frames $\hat{\mathcal{S}}_T = (\hat{s}_1, \hat{s}_2, \dots, \hat{s}_N)$ from different viewpoints; *ii)* force the student to mimic the teacher output starting from a subset $\hat{\mathcal{S}}_S = (\hat{s}_1, \hat{s}_2, \dots, \hat{s}_M) \subset \hat{\mathcal{S}}_T$ with cardinality $M < N$ (in our experiments, $M = 2$ and $N = 8$). The frames in $\hat{\mathcal{S}}_S$ are uniformly sampled from $\hat{\mathcal{S}}_T$ without replacement. This asymmetry between the teacher and the student leads to a self-distillation objective, where the latter can achieve better solutions despite inheriting the same architecture of the former.

To accomplish this, VKD exploits the Knowledge Distillation loss [63]:

$$\mathcal{L}_{\text{KD}} = \tau^2 \, \text{KL}(\boldsymbol{y}_T \parallel \boldsymbol{y}_S) \tag{5.3}$$

where $\boldsymbol{y}_T = \text{softmax}(\boldsymbol{h}_T/\tau)$ and $\boldsymbol{y}_S = \text{softmax}(\boldsymbol{h}_S/\tau)$ are the distributions – smoothed by a temperature τ – we attempt to match[2]. Since the student experiences a different task from the teacher one, Eq. 5.3 resembles the regularisation term imposed by [101] to relieve *catastrophic forgetting*. In a similar vein, we intend to *strengthen* the model in the presence of few images, whilst not *deteriorating* the capabilities it achieved with longer sequences.

In addition to fitting the output distribution of the teacher (Eq. 5.3), our proposal devises additional constraints on the embedding space learnt by the student. In details, VKD encourages the student to mirror the pairwise distances spanned by the teacher. Indicating with $\mathcal{D}_T[i, j] \equiv \mathcal{D}(\mathcal{F}_{\theta_T}(\hat{\mathcal{S}}_T[i]), \mathcal{F}_{\theta_T}(\hat{\mathcal{S}}_T[j]))$ the distance induced by the teacher between the i-th and j-th sets (the same notation $\mathcal{D}_S[i, j]$ also holds for the student), VKD seeks to minimise:

$$\mathcal{L}_{\text{DP}} = \sum_{(i,j) \in \binom{B}{2}} (\mathcal{D}_T[i, j] - \mathcal{D}_S[i, j])^2, \tag{5.4}$$

where B equals the batch size. Since the teacher has access to several viewpoints, we argue that distances spanned in its space yield a powerful description of corresponding identities. From the student perspective, distances preservation provides additional semantic knowledge. Therefore, this holds an effective supervision signal, whose optimisation is made more challenging since fewer images are available to the student.

Even though VKD focuses on *self-distillation*, we highlight that both $\mathcal{L}_{\text{KD}}$ and $\mathcal{L}_{\text{DP}}$ allow to match models with different embedding size, which would not be viable under the minimisation performed by [53]. As an example, it is still possible to distill ResNet-101 ($D = 2048$) into MobileNet-V2 [162] ($D = 1280$).

[2]Since the teacher parameters are fixed, its entropy is constant and the objective of Eq. 5.3 reduces to the cross-entropy between $\boldsymbol{y}_T$ and $\boldsymbol{y}_S$.

Student optimisation The VKD overall objective combines the distillation terms ($\mathcal{L}_{\text{KD}}$ and $\mathcal{L}_{\text{DP}}$) with the ones optimised by the teacher - $\mathcal{L}_{\text{CE}}$ and $\mathcal{L}_{\text{TR}}$ - that promote higher conditional likelihood w.r.t. ground truth labels. To sum up, VKD aims at strengthening the features of a CNN in Re-ID settings through the following optimisation problem:

$$\underset{\theta_S}{\text{argmin}} \quad \mathcal{L}_{\text{VKD}} \equiv \mathcal{L}_{\text{CE}} + \mathcal{L}_{\text{TR}} + \alpha\mathcal{L}_{\text{KD}} + \beta\mathcal{L}_{\text{DP}}, \tag{5.5}$$

where α and β are two hyperparameters balancing the contributions to the total loss $\mathcal{L}_{\text{VKD}}$. We conclude with a final note on the student initialisation: we empirically found beneficial to start from the teacher weights θ_T except for the last convolutional block, which is reinitialised according to the ImageNet pretraining. We argue this represents a good compromise between exploring new configurations and exploiting the abilities already achieved by the teacher.

5.3 Experiments

Evaluation Protocols We indicate the query-gallery matching as x2x, where both x terms are features that can be generated by either a single (I) or multiple frames (V). In the **Image-to-Image (I2I)** setting features extracted from a query set image are matched against features from individual images in the gallery. This protocol has a light impact in terms of resources footprint: however, a single view of the identity may not be enough for identities exhibiting multi-modal distributions. Contrarily, the **Video-to-Video (V2V)** setting enables to capture and combine different modes in the input, but with a significant increase in the number of operations and memory. Finally, the **Image-to-Video (I2V)** setting [225, 226, 112, 202, 114] represents a good compromise: building the gallery may be slow, but it is often performed offline. Moreover, matchings perform extremely fast, as a query comprise only a single image. We remark that *i)* We adopt the standard "*Cross Camera Validation*" protocol, not considering examples of the gallery from the same camera of the query at evaluation and *ii)* even if VKD relies on frames from different camera during train, we strictly adhere to the common schema and switch to tracklet-based inputs at evaluation time.

Evaluation Metrics While settings vary between different dataset, evaluation metrics for Re-Identification are shared by the vast majority of works in the field. We report performance in terms of top-k accuracy and Mean Average Precision (mAP), thus evaluating both recognition accuracy and ranking performance.

5.3.1 Datasets

Person Re-ID **MARS** [221] comprises 19680 tracklets from 6 different cameras, capturing 1260 different identities (split between 625 for the training set, 626 for the gallery and 622 for the query) with 59 frames per tracklet on average. MARS has been automatically annotated, thus leading to errors and false detections [222]. The **Duke** [152] dataset was first introduced for multi-target and multi-camera surveillance purposes, and then expanded to include person attributes and identities (414 ones). Consistently with [53, 173, 107, 129], we use the **Duke-Video-ReID** [203] variant, where identities have been manually annotated from tracking information[3]. It comprises 5534 video tracklets from 8 different cameras, with 167 frames per tracklet on average. Following [53], we extract the first frame of every tracklet when testing in the I2V setting, for both MARS and Duke.

Vehicle Re-ID **VeRi-776** [113] has been collected from 20 fixed cameras, capturing vehicles moving on a circular road in a 1.0 km^2 area. It contains 18397 tracklets with an average number of 6 frames per tracklet, capturing 775 identities split between train (575) and gallery (200). The query set shares identities consistently with the gallery, but differently from the other two sets it includes only a single image for each couple (id, camera). Consequently, all recent methods perform the evaluation following the I2V setting.

Animal Re-ID The **Amur Tiger** [98] Re-Identification in the Wild (ATRW) is a dataset collected from a diverse set of wild zoos. The training set includes 107 subjects and 17.6 images on average per identity; no information is provided to aggregate images into tracklets. It is possible to evaluate only the I2I setting through a remote http server. As in [106], we horizontally flip the training images to double the number of identities available, thus resulting in 214 training identities.

Implementation details Following [60, 107] we adopt the following hyperparameters for MARS and Duke: *i)* each batch contains $P = 8$ identities with $K = 4$ samples each; *ii)* each sample comprises 8 images equally spaced in a tracklet. Differently, for image-based datasets (ATRW and VeRi-776) we increase P to 18 and use a single image at a time. All the teacher networks are trained for 300 epoch using Adam [82], setting the learning rate to 10^{-4} and multiplying it by

[3] In the following, we refer to Duke-Video-ReID simply as Duke. Another variant of Duke named Duke-ReID exists [153], but it does not come with query tracklets.

	MARS				Duke				VeRi-776			
	I2V		V2V		I2V		V2V		I2I		I2V	
	cmc1	**mAP**	**cmc1**	**mAP**	**cmc1**	**mAP**	**cmc1**	**mAP**	**cmc1**	**mAP**	**cmc1**	**mAP**
ResNet-34	80.8	70.7	86.7	78.0	81.3	78.7	93.5	**91.9**	92.3	70.3	93.8	75.0
ResVKD-34	**82.2**	**73.7**	**87.8**	**79.5**	**83.3**	**80.6**	**93.7**	91.6	**95.3**	**76.0**	**94.8**	**79.0**
ResNet-50	82.2	73.4	87.9	81.1	82.3	80.2	**95.0**	**94.2**	93.5	73.2	93.3	77.9
ResVKD-50	**83.9**	**77.3**	**88.7**	**82.2**	**85.6**	**83.8**	**95.0**	93.4	**95.2**	**79.2**	**95.2**	**82.2**
ResNet-101	82.8	75.0	88.6	81.7	83.8	82.9	**96.0**	**94.7**	94.3	74.3	94.5	78.2
ResVKD-101	**85.9**	**77.6**	**89.6**	**82.7**	**86.3**	**85.1**	95.4	93.7	**95.5**	**80.6**	**96.1**	**83.3**
ResNet-50bam	82.6	74.1	88.5	81.2	82.5	80.2	94.9	**93.8**	93.3	72.7	93.8	77.1
ResVKD-50bam	**84.3**	**78.1**	**89.4**	**83.1**	**86.2**	**84.5**	**95.2**	93.5	**96.0**	**78.7**	**95.7**	**81.6**
DenseNet-121	82.7	74.3	89.8	81.9	82.9	80.3	93.7	91.7	91.2	69.2	91.8	74.5
DenseVKD-121	**84.0**	**77.1**	**89.8**	**82.8**	**86.5**	**84.1**	**95.4**	**93.5**	**94.3**	**76.2**	**93.8**	**79.8**
MobileNet-V2	78.6	67.9	86.0	77.1	78.1	74.7	93.3	91.6	88.8	64.7	89.8	69.9
MobileVKD-V2	**83.3**	**74.0**	**88.1**	**79.6**	**83.8**	**80.8**	**94.3**	**92.5**	**92.9**	**70.9**	**92.6**	**75.3**

Table 5.1: Self-Distillation results across datasets, settings and architectures.

0.1 every 100 epochs. During the distillation stage, we feed $N = 8$ images to the teacher and $M = 2$ ones (picked at random) to the student. We found beneficial to train the student longer: so, we set the number of epochs to 500 and the learning rate decay steps at 300 and 450. We keep fixed $\tau = 10$ (Eq. 5.3), $\alpha = 10^{-1}$ and $\beta = 10^{-4}$ (Eq. 5.5) in all experiments. To improve generalisation, we apply data augmentation as described in [120]. Finally, we put the teacher in training mode during distillation (consequently, batch normalisation [69] statistics are computed on a batch basis): as observed in [7], this provides more accurate teacher labels.

5.3.2 Experimental Results

Self-Distillation

In this section we show the benefits of self-distillation for person and vehicle re-id. We indicate the teacher with the name of the backbone (*e.g.* ResNet-50) and append "VKD" for its student (*e.g.* ResVKD-50). To validate our ideas, we do not limit the analysis on ResNet-*; contrarily, we test self-distillation on DenseNet-121 [65] and MobileNet-V2 1.0X [162]. Since learning what and where to look represents an appealing property when dealing with Re-ID tasks [48], we additionally conduct experiments on ResNet-50 coupled with Bottleneck Attention

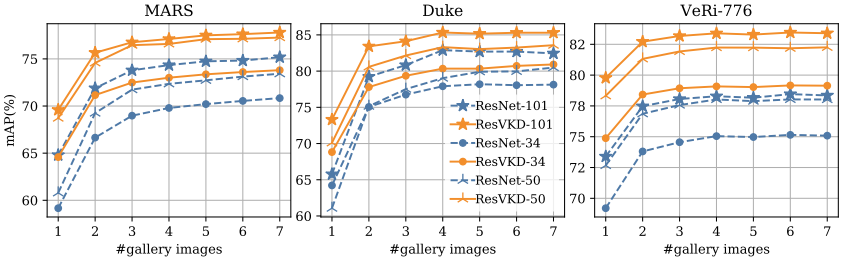


Figure 5.3: Performance (mAP) in the Image-To-Video setting when changing at evaluation time the number of frames in each gallery tracklet.

Modules [140] (ResNet-50bam). Tab. 5.1 reports the comparisons for different backbones: in the vast majority of the settings, *the student outperforms its teacher*. Such a finding is particularly evident when looking at the I2V setting, where the mAP metric gains 4.04% on average. The same holds for the I2I setting on VeRi-776, and in part also on V2V. We draw the following remarks: *i)* in accordance with the objective the student seeks to optimise, our proposal leads to greater improvements when few images are available; *ii)* bridging the gap between I2V and V2V does not imply a significant information loss when more frames are available; on the contrary it sometimes results in superior performance; *iii)* the previous considerations hold true across different architectures. As an additional proof, plots from Fig. 5.3 draw a comparison between models before and after distillation. VKD improves metrics considerably on all three dataset, as highlighted by the bias between the teachers and their corresponding students. Surprisingly, this often applies when comparing lighter students with deeper teachers: as an example, ResVKD-34 scores better than even ResNet-101 on VeRi-776, regardless of the number of images sampled for a gallery tracklet.

Comparison with State-Of-The-Art

Image-To-Video Tables 5.2, 5.3.2 and 5.3.2 report a thorough comparison with current state-of-the-art (SOTA) methods, on MARS, Duke and VeRi-776 respectively. As common practice [53, 11, 146], we focus our analysis on ResNet-50, and in particular on its distilled variants ResVKD-50 and ResVKD-50bam. Our method clearly outperforms other competitors, with an increase in mAP w.r.t. top-scorers of 6.3% on MARS, 8.6% on Duke and 5% on VeRi-776. This results

Method	top$_1$	mAP
P2SNet[200]	55.3	-
Zhang[214]	56.5	-
XQDA[103]	67.2	54.9
TKP[53]	75.6	65.1
STE-NVAN[107]	80.3	68.8
NVAN[107]	80.1	70.2
MGAT[11]	81.1	71.8
ResVKD-50	83.9	77.3
ResVKD-50bam	**84.3**	**78.1**

Table 5.2: MARS **I2V**

Method	top$_1$	mAP
STE-NVAN[107]	42.2	41.3
TKP[53]	77.9	75.9
NVAN[107]	78.4	76.7
ResVKD-50	85.6	83.8
ResVKD-50bam	**86.2**	**84.5**

Table 5.3: Duke **I2V**

Method	top$_1$	mAP
PROVID[114]	76.8	48.5
VFL-LSTM[3]	88.0	59.2
RAM[112]	88.6	61.5
VANet[31]	89.8	66.3
PAMTRI[179]	92.9	71.9
SAN[146]	93.3	72.5
PROVID-BOT[114]	**96.1**	77.2
ResVKD-50	95.2	**82.2**
ResVKD-50bam	95.7	81.6

Table 5.4: VeRi-776 **I2V**

Method	top$_1$	mAP
DuATN[173]	81.2	67.7
TKP[53]	84.0	73.3
CSACSE+OF[30]	86.3	76.1
STA[48]	86.3	80.8
STE-NVAN[107]	88.9	81.2
NVAN[107]	**90.0**	82.8
ResVKD-50	88.7	82.2
ResVKD-50bam	89.4	**83.1**

Table 5.5: MARS **V2V**

Method	top$_1$	mAP
DuATN[173]	81.2	67.7
Matiyali[129]	89.3	88.5
TKP[53]	94.0	91.7
STE-NVAN[107]	95.2	93.5
STA[48]	96.2	94.9
NVAN[107]	**96.3**	**94.9**
ResVKD-50	95.0	93.4
ResVKD-50bam	95.2	93.5

Table 5.6: Duke **V2V**

Method	top$_1$	mAP
PPbM-a [98]	82.5	62.9
PPbM-b [98]	83.3	60.3
NWPU [208]	94.7	75.1
BRL [109]	94.0	77.0
NBU [106]	**95.6**	**81.6**
ResNet-101	92.3	75.7
ResVKD-101	92.0	77.2

Table 5.7: ATRW **I2I**

is totally in line with our goal of conferring robustness when just a single image is provided as query. In doing so, we do not make any task-specific assumption, thus rendering our proposal easily applicable to both person and vehicle Re-ID.

Video-To-Video Analogously, we conduct experiments on the V2V setting and report results in Tab. 5.5 (MARS) and Tab. 5.6 (Duke)[4]. Here, VKD yields the following results: on the one hand, on MARS it pushes a baseline architecture as ResVKD-50 close to NVAN and STE-NVAN [107], the latter being tailored for the V2V setting. Moreover – when exploiting spatial attention modules (ResVKD-50bam) – it establishes new SOTA results, suggesting that a positive transfer occurs when matching tracklets also. On the other hand, the same does not hold true for Duke, where exploiting video features as in STA [48] and NVAN appears rewarding. We leave the investigation of further improvements on V2V to future works. As of today, our proposals is the only one guaranteeing consistent and stable results under both I2V and V2V settings.

[4]Since VeRi-776 does not include any tracklet information in the query set, following all other competitors we limit experiments to the I2V setting only.

	MARS	Duke	VeRi-776
Prior Class.	0.19	0.14	0.06
ResNet-34	**0.61**	**0.73**	**0.55**
ResVKD-34	0.40	0.67	0.51
ResNet-101	**0.71**	**0.72**	**0.73**
ResVKD-101	0.51	0.70	0.68

Table 5.8: Analysis on camera bias, in terms of viewpoint classification accuracy.

In the absence of camera information Here, we address the setting where we do not have access to camera information. As an example, when dealing with animal re-id this information often lacks and datasets come with images and labels solely: can VKD still provide any improvement? We think so, as one can still exploit the visual diversity lying in a bag of randomly sampled images. To demonstrate our claim, we test our proposal on Amur Tigers re-identification (ATRW), which was conceived as an Image-To-Image dataset. During comparisons: *i)* since other works do not conform to a unique backbone, here we opt for ResNet-101; *ii)* as common practice in this benchmark [106, 109, 208], we leverage re-ranking [223]. Tab. 5.3.2 compares VKD against the top scorers in the "Computer Vision for Wildlife Conservation 2019" competition. Importantly, the student ResVKD-101 improves over its teacher (1.5% on mAP and 2.9% on top_5) and places second behind [106], confirming its effectiveness in a challenging scenario. Moreover, we remark that the top-scorer requires additional annotations - such as body parts and pose information - which we do not exploit.

5.4 Model Analysis

VKD reduces the camera bias As pointed out in [182], the appearance encoded by a CNN is heavily affected by external factors surrounding the target object (*e.g.* different backgrounds, viewpoints, illumination ...). In this respect, is our proposal effective for reducing such a bias? To investigate this aspect, we perform a camera classification test on both the teacher (*e.g.* ResNet-34) and the student network (*e.g.* ResVKD-34) by fitting a linear classifier on top of their features, with the aim of predicting the camera the picture is taken from. We freeze all backbone layers and train for 300 epochs (lr $= 10^{-3}$ and halved every 50 epochs). Tab. 5.8 reports performance on the gallery set for different

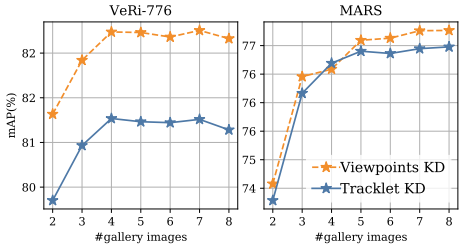


Figure 5.4: Comparison between time and viewpoints distillation.

	Input Bags	MARS I2V		MARS V2V		Duke I2V		Duke V2V	
		cmc1	**mAP**	**cmc1**	**mAP**	**cmc1**	**mAP**	**cmc1**	**mAP**
ResNet-50	$\text{Viewpoints}_{N=2}$	80.1	71.2	84.7	77.0	77.2	75.2	89.2	87.7
ResNet-50	$\text{Tracklets}_{N=2}$	82.3	73.7	87.3	79.9	81.8	80.3	93.7	92.9
ResVKD-50	$\text{Viewpoints}_{N=2}$	**83.9**	**77.3**	**88.7**	**82.2**	**85.6**	**83.8**	**95.0**	**93.4**

Table 5.9: Analysis on different modalities for training the teacher.

teachers and students. To provide a better understanding, we include a baseline that computes predictions by sampling from the cameras prior distribution. As expected: *i)* the teacher outperforms the baseline, suggesting it is in fact biased towards background conditions; *ii)* the student consistently reduces the bias, confirming VKD encourages the student to focus on identities features and drops viewpoint-specific information.

Distilling viewpoints *vs* time Fig. 5.4 shows results of distilling knowledge from multiple views against time (*i.e.* multiple frames from a tracklet). On one side, as multiple views hold more "*visual variety*", the student builds a more invariant representation for the identity. On the opposite, a student trained with tracklets still considerably outperforms the teacher. This shows that, albeit the visual variety is reduced, our distillation approach still successfully exploits it.

Can performance of the student be obtained without distillation? To highlight the advantages of the two-stage procedure above discussed, we here consider a teacher (ResNet-50) trained straightly using few frames ($N = 2$) only. First two rows of Tab. 5.4 show the performance achieved by this baseline (using tracklets

Student	Teacher (#params)	MARS I2V		Duke I2V		VeRi-776 I2V	
		cmc1	**mAP**	**cmc1**	**mAP**	**cmc1**	**mAP**
ResNet-34	ResNet-34 (21.2M)	82.2	73.7	83.3	80.6	94.8	79.0
	ResNet-50 (23.5M)	83.1	75.5	84.1	82.6	**95.1**	80.1
	ResNet-101 (42.5M)	**83.4**	**75.5**	**85.8**	**83.7**	94.9	**80.4**
ResNet-50	ResNet-50 (23.5M)	83.9	77.3	85.6	83.8	95.2	82.2
	ResNet-101 (42.5M)	**84.5**	**77.5**	**85.9**	**84.3**	**95.4**	**83.0**
MobileNet-V2	MobileNet-V2 (2.2M)	83.3	74.0	**83.8**	80.8	92.6	75.3
	ResNet-101 (42.5M)	**83.4**	**74.7**	**83.8**	**81.4**	**93.0**	**76.4**

Table 5.10: Measuring the benefit of VKD for cross-architecture transfer.

and views respectively). Results show that major improvements come from the teacher-student paradigm we devise (third row), instead of simply reducing the number of input images available to the teacher.

Cross-Distillation Differently from other approaches [18, 53], VKD is not confined to self-distillation but instead allows *cross-distillation i.e.* the transfer from a complex architecture (*e.g.*, ResNet-101) into a simpler one (*e.g.*, MobileNet-V2). Here, drawing inspirations from the model compression area, we attempt to reduce the network complexity but, at the same time, increase the profit we already achieve through self-distillation. Tab. 5.10 shows results of cross-distillation, for various combinations of a teacher and a student. It appears that *better the teacher, better the student*: as an example, ResVKD-34 gains an additional 3% mAP on Duke when educated by ResNet-101 rather than "itself".

Student explanation To further assess the differences between teachers and students, we leverage GradCam [169] to highlight the input regions that have been considered paramount for predicting the identity. Fig. 5.5 depicts the impact of VKD for various examples from MARS, VeRi-776 and ATRW. In general, the student network pays more attention to the subject of interest compared to its teacher. For person and animal Re-ID, background features are suppressed (third and last columns) while attention tends to spread to the whole subject (first and fourth columns). When dealing with vehicle Re-ID, one can appreciate how the attention becomes equally distributed on symmetric parts, such as front and rear lights (second, seventh and last columns).

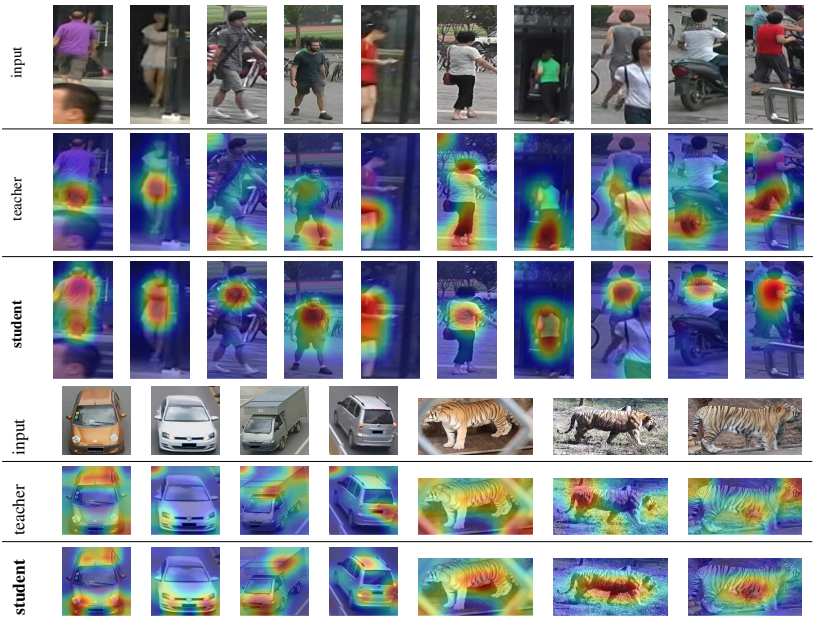


Figure 5.5: Model explanation via GradCam[169] on ResNet-50 (teacher) and ResVKD-50 (student). The student favours visual details characterising the target, discarding external and uninformative patterns.

5.5 Conclusion

An effective Re-ID method requires visual descriptors robust to changes in both background appearances and viewpoints. Moreover, its effectiveness should be ensured even for queries composed of a single image. To accomplish these, the approach discussed in this chapter, called Views Knowledge Distillation (VKD), consists of a teacher-student architecture where the student observes only a small subset of input views. This strategy encourages the student to discover better representations: as a result, it outperforms its teacher at the end of the training. Importantly, VKD shows robustness on diverse domains (person, vehicle and animal), surpassing by a wide margin the state of the art in I2V. The extensive experimental analysis reported in this chapter highlights that the student presents stronger focus on the target and reduces the camera bias.

Chapter 6

Land-Cover Classification

Over the last decades, Remote Sensing has become an enabling factor for a broad spectrum of applications such as disaster prevention [167], wildfire detection [47], vector-borne disease [70], and climate change [155]. These applications benefit from a higher number of satellite imagery captured at unprecedented rhythms [43], making every aspect of the Earth's surface constantly monitored. Machine learning and Computer Vision provide valid tools to exploit these data in an efficient way. Indeed, a synergy between Earth Observation and Deep Learning techniques led to promising results, as highlighted by recent advances in land use and land cover classification [74], image fusion [128], and semantic segmentation [172].

Despite the amount of raw information being significant, the exploitation of these data still raises an open problem. Indeed, the prevailing learning paradigm – the supervised one – frames the presence of labeled data as a crucial factor. However, acquiring a huge amount of ground truth data is expensive and requires expert staff, equipment, and in-field measurements. This often restrains the development of many downstream tasks that are important for paving the way to the above-mentioned applications.

To mitigate such a problem, a common solution [66] exploits models that are pre-trained on the ImageNet [38] dataset. In detail, the learning phase is conducted as follows: firstly, a deep network is trained on ImageNet until it reaches good performance on image categorization; secondly, a fine-tuning step is carried out on a target task (*e.g.* land cover classification). This way, one can achieve acceptable results even in the presence of few labeled examples, as the second step just adapts a set of general-purpose features to the new task. However, this approach is limited

only to the tasks involving RGB images as input. Satellite imagery represents a domain that is quite different from the RGB one, thus making the ImageNet pre-training only partially suitable.

These considerations reveal the need for novel approaches that are tailored for satellite imagery. To build transferable representations, two kinds of approaches arise from the literature: annotation-based methods and self-supervised ones. The authors of [134] fulfill the principle of the first branch by investigating in-domain representation learning. They shift the pre-training stage from ImageNet to a labeled dataset specific for remote sensing. As an example, one could leverage BigEarthNet [177], which has been recently released for land-cover classification. On the other hand, Tile2Vec [73] extracts informative features in a self-supervised fashion. The authors rely on the assumption that spatially close tiles share similar information: therefore, their corresponding representations should be placed closer than tiles that are far apart. In doing so, one does not need labeled data for extracting representations, but lacks robustness when close tiles are not similar.

Similarly to [73], we discuss in the following a representation learning procedure for satellite imagery, which devises a self-supervised algorithm. In more detail, we require the network to recover the RGB information by means of other spectral bands solely. For the rest of the chapter, we adopt the term "spectral bands" for indicating the subset of the bands not including the RGB. Our approach closely relates to colorization, which turns out to encourage robust and high-level feature representations [95, 217]. We feel this pretext task being particularly useful for satellite imagery, as the connection between colors and semantics appears strong: for instance, sea waters feature the blue color, vegetation regions the green one or arable lands prefer warm tones. We inject such a prior knowledge through an encoder-decoder architecture that – differently from concurrent works – exploits spectral bands (*e.g.* short-wave infrared, near-infrared, etc.) instead of grayscale information to infer color channels. Once the model has reached good capabilities on tile colorization, we use its encoder as a feature extractor for the later step, namely fine-tuning on a remote sensing task. We found that the representations learnt by colorization leads to remarkable results and semantically diverge from the ones computed on top of RGB channels. Taking advantage of these findings, we set up an ensemble model, which averages the predictions from two distinct branches at inference time (the one fed with spectral bands, the other with RGB information). We show that ensembling features this way leads to better results. To the best of our knowledge, our work is the first investigating colorization as a guide towards suitable features for remote sensing applications.

To show the effectiveness of our proposal, we assess it in two different set-

tings. Firstly, we conduct experiments on land-cover classification, comparing our solution with two baselines, namely training from scratch and fine-tuning the ImageNet pre-training. We show that colorization is particularly effective when few annotations are available for the target tasks. This makes our proposal viable for scenarios where gathering many labeled data is not practicable. To demonstrate such a claim, we additionally conduct experiments on the "West Nile Virus" cases collected in the frame of the Surveillance plan put in place by the Ministry of Health, with the aim of predicting presence/absence across the Italian territory.

6.1 Preliminaries

6.1.1 Land Cover - Land Use Classification

Recently, the categorization of land-covers has attracted wide interest, as it allows for the collection of statistics, activities planning, and climate changes monitoring. To address these challenges, the authors of [125] exploit Convolutional Neural Networks (CNN) to extract representations encoding both spectral and spatial information. To speed up the learning process, they advocate for a prior dimensionality reduction step across the spectra, as they observe a high correlation in this dimension. Among works focusing on how to exploit spectral bands, [55] devises Recurrent Neural Networks (RNNs) to handle the redundancy underlying adjacent spectral channels. Similarly, [100] proposes a 3D-CNN framework, which can naturally joint spatial and spectral information in an end-to-end fashion without requiring any pre-processing step.

While these approaches concern the design of the feature extractor, our work is primarily engaged in the scenarios in which few labeled examples are available. In these contexts, fine-tuning pre-trained models often mitigate the lack of a large annotated dataset, yielding great performance in some cases [127, 137]. Intuitively, the representations learned from ImageNet (1 million images belonging to 1000 classes) encode a prior knowledge on natural images, thus facilitating the transfer to different domains. Instead, [134] proposes in-domain fine-tuning, where the pre-training stage performs on a remote sensing dataset. The authors found in-domain representations to be effective with limited data (1000 training examples), surpassing the performance given by the ImageNet initialization. Finally, one could reduce overfitting through data augmentation [209] (*i.e.* flip, translation, and rotation), which affects both the diversity and volume of training data.

6.1.2 Unsupervised Representations Learning

Unsupervised and self-supervised methods were introduced to learn general visual features from unlabeled data [75]. These approaches often rely on *pretext tasks*, which attempt to compensate for the lack of labels through an artificial supervision signal. In so doing, the learned representations hopefully embody meaningful information that is beneficial to downstream tasks.

Reconstructions-based methods Under this perspective, generative models can be considered as self-supervised methods, where the reconstruction of the input acts as a pretext task. Denoising Autoencoders [196] contribute to this line of research: here, the learner has to recover the original input from a corrupted version. The idea is that good representations are those capturing stable patterns, which should be recovered even in the presence of a partial or noisy observation. In remote sensing, autoencoders are often applied [104, 123, 125] to reduce the dimensionality of the feature space. This yields the twofold advantage of decreasing the correlation lying in spectral bands and reducing the computational effort.

Classification-based methods [51] frames the pretext task as a classification problem, where the learner guesses which rotation ($0°$, $90°$, $180°$ and $270°$) has been applied to its input. The authors observe that recognizing the transformation behaves as a proxy for object recognition: the higher the accuracy on the upstream task, the higher the accuracy on the downstream one. Considering two random patches from a given image, [41] asks the network to infer the relative position between those. This encourages the learner to recognize the parts that make up the object as well as their relations. Similarly, [138] presents a puzzle to the network, which has to place the shuffled patches back to their original locations.

Colorization-based methods Given a grey-scale image as input, colorization is the process of predicting realistic colors as output. A qualitative analysis conducted in [95] shows that colorization-driven representations capture semantic information, grouping together high-level objects that display low-level variations (*e.g.* color or pose). [40] concerns the ambiguity and ill-posedness of colorization, arguing that several solutions may be assessed for a given grey-scale image. On this basis, the authors exploit Conditional Variational Autoencoder (CVAE) to produce diverse colorizations, thus naturally complying with the multi-modal nature of the problem. Instead, [94] focuses on the design of the inference pipeline

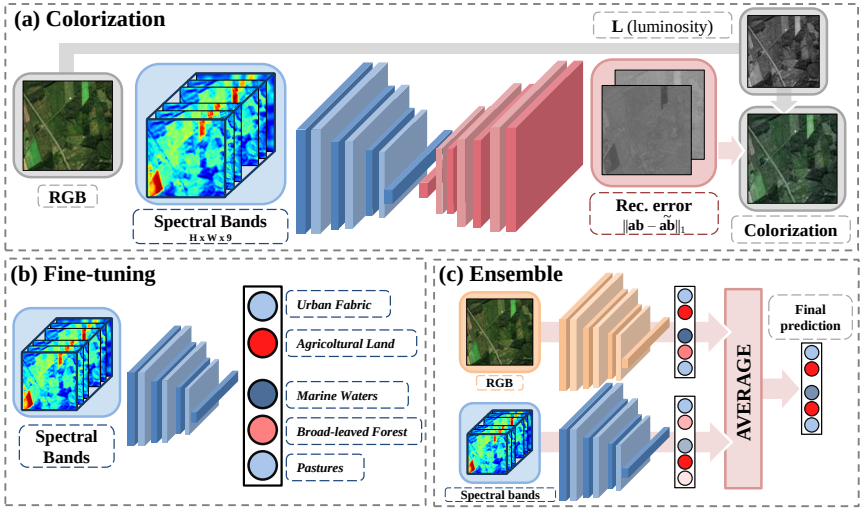


Figure 6.1: An overview of the proposed pipeline.

and proposes a two-stage procedure: *i)* a pixel-wise descriptor is built by VGG-16 feature maps taken at different resolutions; *ii)* the descriptors are then fed into a fully connected layer, which outputs hue and chroma distributions. Split-Brain Autoencoders [217] relies on a network composed of two disjoint modules, each of which predicts a subset of color channels from another. The authors argue that this schema induces transferable representations, the latter taking into account all input dimensions (instead of gray-scale solely).

6.2 Proposed Approach

Overview Our main goal consists in finding a good initialization for the classifier, in such a way that it can later capture meaningful and robust patterns even in presence of few labeled data. To this purpose, we devise a two-stage procedure tailored for satellite imagery tasks, which prepends a colorization step (Sec. 6.2.1) to a fine-tuning one (Sec. 6.2.2).

As depicted in Fig. 6.1 (a), our proposal leverages an encoder-decoder architecture for feature learning. In doing so, we do not require the model to reconstruct

its input: differently, we set up an asymmetry between input (spectral bands) and output (color channels). This way, we expect the encoder to capture meaningful information about soil and environmental characteristics. Afterward, we exploit the encoder and its representation capabilities to tackle a downstream task (*e.g.* land cover classification, see Fig. 6.1 (b)). Eventually, an ensemble model (see Sec. 6.2.3 for additional details) further refines the final prediction combining the outputs from the two input modalities (RGB and spectral bands).

6.2.1 Colorization

In formal terms, the encoder network $\mathcal{F}$ takes $\mathbf{S} \in \mathbb{R}^{H \times W \times C}$ as input, where C equals the number of spectral bands available to the model and H and W the input resolution (height and width respectively). The decoder network produces a tensor $\widehat{\mathbf{X}} \in \mathbb{R}^{H \times W \times 2}$, which yields the pixel-wise predictions in terms of a and b coordinates in the CIE *Lab* color space. On this latter point, a naive strategy would simply define the expected output in terms of RGB: nevertheless, as pointed out in [94], modeling colors as RGB values may not yield an effective training signal. Differently, we adhere to the guideline described in [216] and frame the problem in the CIE *Lab* space. Here, a color is defined with a lightness component L and $a * b$ values carrying the chromatic content. The effectiveness of this space comes from the fact that colors are encoded accordingly to human perception: namely, the distance between two points reflects the amount of visually perceived change between the corresponding colors.

Encoder We opt for ResNet18 [58] as backbone network for the encoder, which hence consists of four blocks with two residual units each. As pointed out in [87], thanks to their residual units and skip connections, ResNet-based networks are more suitable for self-supervised representation learning. Indeed, when compared to other popular architectures (*e.g.* AlexNet), residual networks favorably preserve representations from degrading towards the end of the network and therefore results in better performance.

Decoder In designing the decoder network, we mirror the architecture of the encoder, replacing the first convolutional layer of each residual block with its transposed counterpart. Moreover, we add an upsampling operation to the top of the decoder, followed by a batch normalization layer, a ReLU activation, and a transposed convolution. The latter reduces the number of features maps to 2: this way, the output dimensionality matches the ground truth one.

Colorization Loss Recent works [217, 216, 95] investigate various loss functions, questioning their contributions to colorization results (intended as performance on either the target task or the pretext one). Despite a regression objective (*e.g.* the mean squared error) being a valid baseline, these works show that treating the problem as a multinomial classification leads to better results. However, the overall training time increases considerably because of the additional information taken into account. In our case, this would add up to the burdensome computations required by hyperspectral images, thus resulting even more expensive. For this reason, we limit our experiments to the mean absolute error $\mathcal{L}_1(\cdot, \cdot)$, as follows:

$$\mathcal{L}_1(\widehat{\mathbf{X}}, \mathbf{X}) = \lambda \sum_{h,w} \left| \widehat{x}_{h,w}^{\,(a)} - x_{h,w}^{(a)} \right| + \left| \widehat{x}_{h,w}^{\,(b)} - x_{h,w}^{(b)} \right|, \tag{6.1}$$

where $\mathbf{X}$ represents the $a * b$ ground truth colorization and $\lambda = 100$ is a weighting term that prevents numerical instabilities.

6.2.2 Fine-tuning

Once the encoder-decoder has been trained, we turn our attention to the downstream task exploiting the encoder $\mathcal{F}(\cdot)$ as pre-trained feature extractor. We need just a single amendment to the network: a final linear transformation that maps bottleneck features $\mathbf{H} = \mathcal{F}(\mathbf{S})$ to the classification output space $\widehat{\mathbf{y}} = \mathbf{W}^T\mathbf{H} + \mathbf{b}$.

Classification Loss We make use of two different losses in our experiments: when dealing with a multi-label task as the land cover classification one (*i.e.* each example can be categorized into multiple classes), the objective function resembles a binary cross-entropy term averaged over C classes:

$$\mathcal{L}(\widehat{\mathbf{y}}, \mathbf{y}) = -\frac{1}{C} \sum_i \mathbf{y}_i \log \sigma\left(\widehat{\mathbf{y}}_i\right) + \left(1 - \mathbf{y}_i\right) \log\left(1 - \sigma\left(\widehat{\mathbf{y}}_i\right)\right),$$

where $\mathbf{y}$ is the ground-truth multi-hot encoding vector and σ the sigmoid function. Differently, we use the binary cross-entropy for the West Nile Disease case study.

6.2.3 Model Ensemble

As pointed out in [217], a network trained on colorization specializes just on a subset of the available data (in our case, spectral bands) and cannot exploit the information coming from its ground truth (the RGB color images). To further

take advantage of color information, we set up an ensemble model at inference time (so, no additional training steps required). As shown in Fig. 5.2 (c), the ensemble is formed by two independent branches taking the RGB channels and the spectral bands as input respectively. The first one is pre-trained on classification (ImageNet) and the second one on colorization; both are fine-tuned separately on the given classification task. The ensemble-level predictions are simply computed by averaging the responses from the two branches $\widehat{\mathbf{y}}_{\text{ENS}} = {}^{\sigma(\widehat{\mathbf{y}}_{\text{RGB}})+\sigma(\widehat{\mathbf{y}}_{\text{SPECTRAL}})}/_{2}$.

6.3 Experiments

In this section, we test our proposal as a pre-training strategy for the later fine-tuning step. We compare the results yielded by colorization to those achieved by two baselines: training from scratch [57] and the common ImageNet pre-training. In doing so, we mimic scenarios with few labeled data by reducing the amount of examples available at training time (*e.g.* $1\,000$, $5\,000$, etc...).

6.3.1 Datasets

The two datasets we rely on data acquired through the Sentinel-2A and 2B satellites developed by the European Space Agency (ESA). These satellites provide a multi-spectral imagery over the earth with 12 spectral bands (covering the visible, near and short wave infrared part of the electromagnetic spectrum) at three different spatial resolutions (10, 20 and 60 meters per pixel).

Land-cover classification - BigEarthNet In Remote Sensing, the main bottleneck in the adoption of deep networks was the lack of a large training set. Indeed, existing datasets (as Eurosat [59], PatterNet [224], UC Merced Land Use Dataset [207]) include a small number of annotated images, hence resulting inadequate for training very deep networks. To overcome this problem, [177] introduces BigEarthNet, a novel large scale dataset collecting $590\,326$ tiles. Each example comprises of 12 bands (RGB included) and multiple land-cover classes (provided by the CORINE Land Cover (CLC) database [46]) as ground truth.

Originally, the number of classes amounted to 43: but, the authors of [178] argue that some CORINE classes cannot be easily inferred by looking at Sentinel-2 images solely. Indeed, some labels may not be recognizable at such low resolution (the highest one is 120×120 pixels for $10m$ bands) and other ones would require temporal information for being correctly discriminated (*e.g.* non-irrigated arable

land *vs.* permanently irrigated land). For these reasons, in our experiments we adopt the class-nomenclature proposed in [178], which reduces the number of classes to 19. Moreover, we discard the 70 987 patches displaying lands that are fully covered by clouds, cloud shadows, and seasonal snow.

West Nile Disease Dataset Numerous studies have examined the complex interactions among vectors, hosts, and pathogens [70, 184]: one of the major threat worldwide studied is represented by West Nile Disease (WND), a mosquito-borne disease caused by West Nile virus (WNV). Mosquitoes presence and abundance have been extensively proved to be associated with climatic and environmental factors such as temperatures, vegetation, rainfall [184, 19, 35], and remote sensing has been an important key source for data collection. Our ability to store data continues to expand rapidly; this requires new techniques processing Earth Observation (EO) data and establishing pipelines that turn near real-time "big data" into "smart data" [144]. In this context, Deep techniques could provide useful tools to identify patterns able to make accurate predictions of the re-emergence and spread of the West Nile Disease in Italy. With this aim, we collected data from the Copernicus program and paired Sentinel 2 EO data with ground truth WND data.

Disease sites are collected through the National Disease Notification System of the Ministry of Health (SIMAN www.vetinfo.sanita.it) [33]. We start with the analysis of the 2018 epidemic, one of the most spread on the Italian territory. We frame the problem as a binary classification task with the final purpose of predicting positive and negative WND sites analyzing multi-spectral bands. Positive cases are geographically located mainly in Po valley, in Sardinia and some spots in the rest of Italy [150]: the location of each case of birds, mosquitoes and horses, was visually inspected for the accuracy needs in the analysis. Negative sites, being not always available in the national database due to the surveillance plan strategy, were derived as pseudo-absence ground truth data, either in the space (points located in areas where the disease was never reported in the past) and in the time (a random date in months previous the reported positivity in mosquitoes collections).

WND dataset contains 1 488 distinct cases (962 negatives and 526 positives): each case comes with a variable number of Sentinel-2 patches (corresponding to various acquisitions over time), leading to 18 684 images in total.

6.3.2 Evaluation Protocol

Land-Cover Classification We strictly follow the guidelines provided by [134] when assessing the performance on the BigEarthNet benchmark. Namely, we

form the training set by sampling 60% of the total examples considered, retaining 20% for the validation set and 20% for the test set. We measure the results in terms of Mean-Average-Precision (mAP), which also considers the order in which predictions are given to the user. We check the performance every 10 epochs and retain the weights that yield the higher mAP score on the validation set.

West Nile Disease Here, we adopt the stratified holdout strategy, which ensures the class probabilities of training and test being close to each other. In detail, we exploit 80% of the total examples for the training set and the remaining 20% for the test set. The metrics of interest are precision, recall and F1 score, the latter accounting for the slight imbalance that occurs at class level (indeed, negatives cases appear more frequently than positives ones).

Implementation details

BigEarthNet We exploit the normalization technique described in [145, 144] computing the 2^{nd} and 98^{th} percentile values to normalize each band. This method is more robust than the common min-max normalization, as it is less sensitive to outliers. As the spectral bands come at different spatial resolutions, we apply a cubic interpolation to get a dimension of 128×128.

Colorization To broaden the diversity of available data, we apply data augmentation (*i.e.* rotation, horizontal and vertical flip). We train for 50 epochs on the full BigEarthNet, setting the batch size equal to 16 and using Stochastic Gradient Descent (SGD) (with a learning rate fixed at 0.01).

Land-Cover Classification We train the model for 30 epochs whether the full dataset is available; otherwise we increase the epochs to 50. The learning rate is set to 0.1 and divided by 10 at the 10^{th} and 40^{th} epoch. The batch size equals 64.

West Nile Disease Differently from the previous cases, we apply neither upscaling nor pixel-normalization, as all channels are provided at the same resolution (224×224) and their values lie within the range $[0, 1]$. We leverage the network trained for colorization on BigEarthNet. Since we rely on a subset of the spectral bands (B_1, B_{8A}, B_{11} and B_{12}), we fix the first convolutional layer so that it takes 4 channels as input. We optimize the model for 30 epochs, with a batch size of 32 and an initial learning rate of 0.001, multiplied by 0.1 after 25 epochs.

Input	pre-training	1k	5k	10k	50k	Full
RGB	from scratch	.486	.608	.645	.744	.851
RGB	ImageNet	.620	.695	.726	.786	**.879**
Spectral	from scratch	.555	.667	.711	.767	.866
Spectral	ImageNet	.578	.627	.681	.773	**.879**
Spectral	Color. (our)	**.622**	**.730**	**.760**	**.793**	.860
Ensemble	ImagNet+ImageNet	.649	.707	.749	.815	**.904**
Ensemble	Color.+ImageNet	**.656**	**.751**	**.778**	**.823**	.896

Table 6.1: Performance (mAP) on BigEarthNet for different strategies to vary the number of training examples.

Input	pre-training	1k	5k	10k	50k	Full
RGB	ImageNet	.620	.695	.726	.786	.879
Spectral	Colorization	.622	.730	.760	.793	.860
Ensemble	ImagNet+ImageNet	.649	.707	.749	.815	**.904**
Ensemble	Color.+ImageNet	**.656**	**.751**	**.778**	**.823**	.896

Table 6.2: Ensemble model – results (mAP) on BigEarthNet.

6.3.3 Experimental Results

Results of Colorization Based on the performance reported in Tab. 6.1, the initialization offered by colorization surpasses the other alternatives. This holds in presence of scarce data, thus complying with the goals outlined at the beginning of the chapter. This does not apply when the entire training set (519k examples) is available: such evidence – already encountered in [134] – deserves more investigations that we will conduct in future studies. Results shown by Tab. 6.1 let us draw additional remarks: *i)* as one would expect, the ImageNet pre-training performs good for RGB inputs; however, when dealing with the spectral domain, even a random initialization outperforms it; *ii)* colorization is the sole that rewards the exploitation of spectral bands and justifies their usage in place of RGB.

Input	pre-training	acc.	pr.	rc.	F1
Random classifier	-	.503	.391	.395	.393
RGB	from scratch	.652	.542	.941	.688
RGB	ImageNet	.865	.819	.857	.838
$B_{1,8A,11,12}$	from scratch	.756	.662	.817	.732
$B_{1,8A,11,12}$	Colorization	.852	.823	.811	.817
Ensemble	Color.+ImageNet	**.880**	**.855**	**.850**	**.852**

Table 6.3: Performance (acc. accuracy, pr. precision, rc. recall) on the West Nile Disease case study, for different methods and pre-training strategies.

Method	pr.	rc.	F1
K-Branch CNN	.716	**.789**	.727
VGG19	.798	.767	.759
ResNet-50	.813	.774	.771
ResNet-101	.801	.774	.764
ResNet-152	.817	.762	.765
Ensemble (our)	**.843**	.781	**.811**

Table 6.4: Comparison between baselines and our ensemble (BigEarthNet).

Results of the model ensemble We first assess the effectiveness of the ensemble discussed in Sec. 6.2.3 on BigEarthNet. In this regard, Tab. 6.2 compares the performance that can be reached when using a twofold source of information (RGB and spectral bands): firstly, the ensemble model largely outperforms those considering a single modality; secondly, colorization presents an improvement over the ImageNet pre-training.

Tab. 6.3 reports the results achieved on the West Nile Disease case study discussed in Sec 6.3.1. To provide a better understanding, we additionally furnish a simple baseline (*i.e.* "random classifier") that computes predictions by randomly guessing from the class-prior distribution of the training set. As a first remark, all the networks we trained exceed random guessing, hence suggesting they effectively learned meaningful and suitable features for the problem at hand. Secondly, the

ensemble model plays an important role even in this case, surpassing networks based on a single modality by a consistent margin.

Comparison with the state of the art To further highlight the contributions of our proposal, we compare it with the networks discussed in [177]. Results reported in Tab. 6.4 confirm the above intuitions: the ensemble we build upon ResNet-18 outperforms heavier and overparametrized networks like ResNet-101 or ResNet-152. Notably, we found a large improvement in precision, suggesting that our proposal is capable of returning only the categories that are relevant to the semantics of the input tile.

It is noted the fairness of the comparisons above, as both our ensemble and the baselines leverage the same amount of information in input (namely, spectral bands and color channels). Nevertheless, an important difference subsists in the way information is consumed: while [177] stacks both the input modalities to form a single input tensor, we distinguish two independent paths that eventually cross in the output space. This way, we can benefit from two different pre-training, each one being devoted to its modality: the one offered by colorization – which works well for spectral bands – and the ImageNet one – which instead represents a natural and reasonable choice for dealing with RGB images.

6.4 Model Analysis

Towards diverse feature sets We believe the strength of our ensemble approach being a result of the diversity among the individual learners. We investigate the truthfulness of such a claim from a *model explanation* perspective, questioning which information in the input makes our models arrive at their decisions [161]. In particular, we take advantage of GradCam [169] to assess whether the two branches look for different properties within their inputs. The third and fourth rows of Fig. 6.2 highlight the input regions that have been considered important for predicting the target category (we limit the analysis to the class denoting the highest confidence score). As one can see, the explanations provided by the two branches visually diverge, thus qualitatively confirming the weak correlation between their representations.

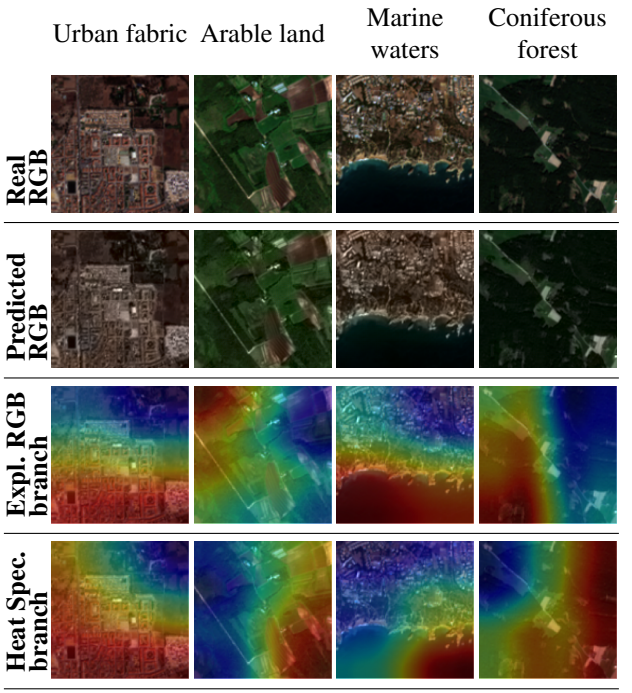


Figure 6.2: Examples of the BigEarthNet dataset, the predicted colorizations and visual explanations provided by the ensemble method for RGB and spectral inputs.

6.5 Conclusion

This chapter has discussed a self-supervised approach for satellite imagery, which moves towards a proper initialization for Remote Sensing tasks. It builds upon two steps: firstly, we ask an encoder-decoder architecture to predict color channels from those capturing spectral information (colorization); secondly, we exploit its encoder as a pre-trained feature extractor for a classification task (*e.g.* land-cover categorization). We have observed that the proposed initialization leads to remarkable results, exceeding the baselines in presence of scarce labeled data. Moreover, the representations learned through colorization are different from the ones driven by the RGB channels. Based on this finding, we have set up an ensemble model that achieves the highest results in the scenarios under consideration.

Chapter 7

Conclusions

This thesis has explored several widespread Computer Vision tasks, highlighting how modern Deep Learning techniques can pave the way towards impressive advances in terms of performance. This has been made possible by outstanding and continuous developments that occurred in the last decade: essentially, researchers and practitioners can now benefit from both more powerful hardware resources [9] and larger datasets [39]. However, these assets can be more profitably exploited by exploiting the principled and insightful approaches presented in each chapter of this thesis. In particular, we have extensively shown that the incorporation of prior beliefs (*i.e.* a set of reasonable guesses about the phenomenon under examination) into the learning process leads to deep learning models characterized by higher generalization capabilities. In this regard, the author feels that the value of this work lies in the diversity and heterogeneity of the domains and fields covered during these three years. Remarkably, it has been shown that a common and longstanding design principle – such as the incorporation of prior knowledge – can be effectively applied to disparate contexts, featuring diverse input modalities, tasks and architectures. To provide a brief summary:

- Chapter 2 deals with graph classification and points out that a tailored layer exploiting the underlying geometrical structure provides better classification results.
- Chapter 3 proposes a framework for novelty detection in which prior knowledge is embodied by an auxiliary component acting in latent space.

- Chapter 4 remarks that *catastrophic forgetting* can be strongly mitigated by using prior model's responses while learning new tasks.
- Chapter 5 shows that prior information can be imposed as a reasonable property in terms of feature representation (*i.e.* being able to recover many visual details from a few views of the target object).
- Chapter 6 finally highlights network pretraining as a further convenient way to incorporate a preliminary guess on the usage of a land territory depicted in an image.

As a final note, the author would like to report a very recent trend regarding the design of deep architectures, which seems to point towards a slightly different direction. Indeed, Vision Transformers [42] – which have recently arisen as promising architectures for computer vision tasks – relax the rigid interaction pattern of 2D-convolutions (*i.e.*, locality and weight sharing) and, instead, exploit a self-attention mechanism across embeddings of patches of pixels (thus allowing for both global interactions and flexible spatial relations). Such an increased degree of freedom has proven to be especially rewarding for scenarios featuring large datasets (in the order of 14M-300M images); otherwise, the small-data regime still promotes those architectures incorporating the above-mentioned convolutional constraints. In light of this, we feel that future works should strive for new advances in the design of the self-attention mechanism, investigating how soft prior beliefs can be introduced without involving restrictions to the learning process.

Appendix A

List of publications

Statement of contributions

In the following, the author of this thesis outlines his contributions to each covered topic:

- **Graph Classification**. The research presented in Chapter 2 has been also published in [142], which is the first peer-reviewed article by the author. In this work, the author delved firsthand into the study of graph convolutions, with particular focus on pooling techniques tailored for graph signals. Therefore, he firstly formalized a preliminary mathematical background and then developed an approach on top of it. Eventually, he set up the entire experimental evaluation. The other authors involved in the publications gave a valuable contribution in terms of writing, as well as ideas and insights to extend the analysis with meaningful ablation studies.

- **Novelty Detection**. The publication [1] underlying Chapter 3 is the second article the author worked on. Even though he is not the lead author of the research, he gave a valuable contribution, consisting of the following points: *i)* he shaped the mathematical modeling of the original idea; *ii)* he provided insightful suggestions to improve the model performance; *iii)* several ablation studies and qualitative experiments originated from his intuitions; *iiii)* he heavily contributed to the final draft of the article.

- **Continual Learning**. Chapter 4 is based on [23]. Similar to the previous chapter, the author furthered the research in both theoretical and experi-

mental terms. In particular, he spent his efforts principally on Sec. 4.4, which explains the effectiveness of the proposed approach from multiple points of view.

- **Re-Identification** The teacher-student paradigm introduced in [143], which is the source for Chapter 4, came from an idea of the author of this thesis. Following his intuition, he developed the approach from scratch and then experimentally compared it with other existing methods. Importantly, the author considers this work worth mentioning, as it allowed him to delve deeply into the topic of Knowledge Distillation, which has become a great source of inspiration for several subsequent works [23, 20].
- **Land-Cover Classification**. Finally, the author was involved in [197]: he oversaw both the development and experimental validation of the proposed method, giving as well an important contribution in terms of writing.

List of publications

The following list of publications includes all conference papers, journal articles, and book chapters published during my Ph.D. period, as well as recent pre-prints. Content and experimental results published in some of these papers have been included in the previous chapters, with explicit permission given by the other authors.

[1] Angelo Porrello, Luca Bergamini, and Simone Calderara. Robust re-identification by multiple views knowledge distillation. In *Proceedings of the European Conference on Computer Vision*. Springer, 2020. 3, 86

[2] Angelo Porrello, Davide Abati, Simone Calderara, and Rita Cucchiara. Classifying signals on irregular domains via convolutional cluster pooling. In *International Conference on Artificial Intelligence and Statistics*. PMLR, 2019. 2, 85

[3] Davide Abati, Angelo Porrello, Simone Calderara, and Rita Cucchiara. Latent space autoregression for novelty detection. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2019. 2, 85

[4] Pietro Buzzega, Matteo Boschini, Angelo Porrello, Davide Abati, and Simone Calderara. Dark experience for general continual learning: a strong, simple baseline. In *Advances in Neural Information Processing Systems*, 2020. 3, 85, 86

[5] Pietro Buzzega, Matteo Boschini, Angelo Porrello, and Simone Calderara. Rethinking experience replay: a bag of tricks for continual learning. In *International Conference on Pattern Recognition*. IEEE, 2021.

[6] Stefano Vincenzi, Angelo Porrello, Pietro Buzzega, Marco Cipriano, Pietro Fronte, Roberto Cuccu, Carla Ippoliti, Annamaria Conte, and Simone Calderara. The color out of space: learning self-supervised representations for earth observation imagery. In *International Conference on Pattern Recognition*. IEEE, 2021. 3, 86

[7] Angelo Porrello, Stefano Vincenzi, Pietro Buzzega, Simone Calderara, Annamaria Conte, Carla Ippoliti, Luca Candeloro, Alessio Di Lorenzo, and Andrea Capobianco Dondona. Spotting insects from satellites: modeling the presence of culicoides imicola through deep cnns. In *International Conference on Signal-Image Technology & Internet-Based Systems (SITIS)*. IEEE, 2019.

[8] Luca Bergamini, Angelo Porrello, Andrea Capobianco Dondona, Ercole Del Negro, Mauro Mattioli, Nicola D'alterio, and Simone Calderara. Multi-views embedding for cattle re-identification. In *International Conference on Signal-Image Technology & Internet-Based Systems*. IEEE, 2018. 54

[9] Luca Candeloro, Carla Ippoliti, Federica Iapaolo, Federica Monaco, Daniela Morelli, Roberto Cuccu, Pietro Fronte, Simone Calderara, Stefano Vincenzi, Angelo Porrello, et al. Predicting wnv circulation in italy using earth observation data and extreme gradient boosting model. *Remote Sensing*, 2020.

[10] Abigail R Trachtman, Luca Bergamini, Andrea Palazzi, Angelo Porrello, Andrea Capobianco Dondona, Ercole Del Negro, Andrea Paolini, Giorgio Vignola, Simone Calderara, and Giuseppe Marruchella. Scoring pleurisy in slaughtered pigs using convolutional neural networks. *Veterinary research*, 2020.

[11] Matteo Boschini, Lorenzo Bonicelli, Pietro Buzzega, Angelo Porrello, and Simone Calderara. Class-incremental continual learning into the extended der-verse. *arXiv preprint arXiv:2201.00766*, 2022. Submitted to IEEE Transactions on Pattern Analysis and Machine Intelligence. 86

Appendix B

Activities carried out during Ph.D.

Teaching activities

- Teaching assistant for the "Pattern Recognition and Machine Learning" graduate course (2018, 2019).
- Lecturer for the training course on Artificial Intelligence, organized by The Istituto Zooprofilattico Sperimentale dell'Abruzzo e del Molise "Giuseppe Caporale" (2018).
- Lecturer for the Master ADBoT – Autonomous Driving and enaBling Technologies – organized by the University of Trento (UNITN) in collaboration with the University of Modena and Reggio Emilia (2019).
- Lecturer for the Short Master in Machine learning e Deep learning, organised by Fondazione Democenter (2018, 2019, 2020).

Participation to national and international projects

- "FAR2016" – Ubiquitous objective measures of intergroup nonverbal behaviors, UBIVNB – financed by the University of Modena and Reggio Emilia

- "AI4VECT" – Artificial Intelligence and Remote Sensing: innovative methods for monitoring vectors and the associated ecological/environmental variables – financed by the Italian Ministry of Health.
- "InSecTT" – Intelligent Secure Trustable Things – funded by the Electronic Component Systems for European Leadership Joint Undertaking under grant agreement 876038.

Reviewing

- International Conference on Computer Vision and Pattern Recognition (CVPR)
- International Conference on Computer Vision (ICCV)
- European Conference on Computer Vision (ECCV)
- Conference on Neural Information Processing Systems (NeurIPS)
- International Joint Conference on Artificial Intelligence (IJCAI)
- International Conference on Pattern Recognition (ICPR)
- ACM Transactions on Multimedia Computing Communications and Applications (TOMM)

Conferences and schools attended

- VISMAC 2018: Summer School On Machine and Vision Intelligence, organized by the International Association for Pattern Recognition (IAPR). Vico Equense, Italy.
- SITIS 2018: 14th International Conference on Signal Image Technology & Internet Based Systems. Las Palmas de Gran Canaria, Spain.
- AISTATS 2019 : The 22nd International Conference on Artificial Intelligence and Statistics. Naha, Okinawa, Japan.
- ICIAP 2019: 20th International Conference on Image Analysis And Processing. Trento, Italy.
- ECCV 2020: 16th European Conference on Computer Vision. Online.

Bibliography

[1] Davide Abati, Angelo Porrello, Simone Calderara, and Rita Cucchiara. Latent space autoregression for novelty detection. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2019. 2, 85

[2] Amit Adam, Ehud Rivlin, Ilan Shimshoni, and Daviv Reinitz. Robust real-time unusual event detection using multiple fixed-location monitors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2008. 22

[3] Saghir Ahmed Saghir Alfasly, Yongjian Hu, Tiancai Liang, Xiaofeng Jin, Qingli Zhao, and Beibei Liu. Variational representation learning for vehicle re-identification. In *IEEE International Conference on Image Processing*, 2019. 64

[4] Rahaf Aljundi, Min Lin, Baptiste Goujaud, and Yoshua Bengio. Gradient based sample selection for online continual learning. In *Advances in Neural Information Processing Systems*, 2019. 39, 40, 41, 46, 47, 49

[5] James Atwood and Don Towsley. Diffusion-convolutional neural networks. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, 2016. 6

[6] Philip Bachman, Ouais Alsharif, and Doina Precup. Learning with pseudo-ensembles. In *Advances in Neural Information Processing Systems*, 2014. 45

[7] Hessam Bagherinezhad, Maxwell Horton, Mohammad Rastegari, and Ali Farhadi. Label refinery: Improving imagenet classification through label progression. *arXiv preprint arXiv:1805.02641*, 2018. 62

[8] Shaojie Bai, J Zico Kolter, and Vladlen Koltun. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. *arXiv preprint arXiv:1803.01271*, 2018. 26

[9] Toru Baji. Evolution of the gpu device widely used in ai and massive parallel processing. In *2018 IEEE 2nd Electron Devices Technology and Manufacturing Conference (EDTM)*. IEEE, 2018. 83

[10] Johannes Ballé, Valero Laparra, and Eero P Simoncelli. End-to-end optimized image compression. *International Conference on Learning Representations*, 2017. 21

[11] Liqiang Bao, Bingpeng Ma, Hong Chang, and Xilin Chen. Masked graph attention network for person re-identification. In *IEEE International Conference on Computer Vision and Pattern Recognition Workshops*, 2019. 55, 63, 64

[12] Andrew Barto, Marco Mirolli, and Gianluca Baldassarre. Novelty or surprise? *Frontiers in psychology*, 2013. 20

[13] Arslan Basharat, Alexei Gritai, and Mubarak Shah. Learning object motion patterns for anomaly detection and improved object detection. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*. IEEE, 2008. 22

[14] Matthias Bauer and Andriy Mnih. Resampled priors for variational autoencoders. *International Conference on Artificial Intelligence and Statistics*, 2019. 22

[15] Suzanna Becker and Geoffrey E Hinton. Self-organizing neural network that discovers surfaces in random-dot stereograms. *Nature*, 1992. 45

[16] Ari S. Benjamin, David Rolnick, and Konrad P. Kording. Measuring and regularizing networks in function space. *International Conference on Learning Representations*, 2019. 40, 43, 45, 46

[17] Luca Bergamini, Angelo Porrello, Andrea Capobianco Dondona, Ercole Del Negro, Mauro Mattioli, Nicola D'alterio, and Simone Calderara. Multi-views embedding for cattle re-identification. In *IEEE International Conference on Signal-Image Technology & Internet-Based Systems*, 2018. 54

[18] Shweta Bhardwaj, Mukundhan Srinivasan, and Mitesh M Khapra. Efficient video classification using fewer frames. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2019. 53, 56, 67

[19] Donal Bisanzio, Mario Giacobini, Luigi Bertolotti, Andrea Mosca, Luca Balbo, Uriel Kitron, and Gonzalo M Vazquez-Prokopec. Spatio-temporal patterns of distribution of west nile virus vectors in eastern piedmont region, italy. *Parasites & vectors*, 2011. 77

[20] Matteo Boschini, Lorenzo Bonicelli, Pietro Buzzega, Angelo Porrello, and Simone Calderara. Class-incremental continual learning into the extended der-verse. *arXiv preprint arXiv:2201.00766*, 2022. Submitted to IEEE Transactions on Pattern Analysis and Machine Intelligence. 86

[21] Jochen Bröcker. Reliability, sufficiency, and the decomposition of proper scores. *Quarterly Journal of the Royal Meteorological Society*, 2009. 51

[22] Joan Bruna, Wojciech Zaremba, Arthur Szlam, and Yann Lecun. Spectral networks and locally connected networks on graphs. In *International Conference on Learning Representations*, 2014. 4, 6, 9

[23] Pietro Buzzega, Matteo Boschini, Angelo Porrello, Davide Abati, and Simone Calderara. Dark experience for general continual learning: a strong, simple baseline. In *Advances in Neural Information Processing Systems*, 2020. 3, 85, 86

[24] Simone Calderara, Uri Heinemann, Andrea Prati, Rita Cucchiara, and Naftali Tishby. Detecting anomalies in people's trajectories using spectral graph analysis. *Computer Vision and Image Understanding*, 2011. 20

[25] Antoni Chan and Nuno Vasconcelos. Ucsd pedestrian database. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2008. 32

[26] Pratik Chaudhari, Anna Choromanska, Stefano Soatto, Yann LeCun, Carlo Baldassi, Christian Borgs, Jennifer Chayes, Levent Sagun, and Riccardo Zecchina. Entropy-sgd: Biasing gradient descent into wide valleys. In *International Conference on Learning Representations*, 2017. 49, 50

[27] Arslan Chaudhry, Puneet K Dokania, Thalaiyasingam Ajanthan, and Philip HS Torr. Riemannian walk for incremental learning: Understanding forgetting and intransigence. In *Proceedings of the European Conference on Computer Vision*, 2018. 40

[28] Arslan Chaudhry, Albert Gordo, Puneet K Dokania, Philip Torr, and David Lopez-Paz. Using Hindsight to Anchor Past Knowledge in Continual Learning. In *AAAI Conference on Artificial Intelligence*, 2021. 39, 40, 41, 46, 47

[29] Arslan Chaudhry, Marc'Aurelio Ranzato, Marcus Rohrbach, and Mohamed Elhoseiny. Efficient lifelong learning with a-gem. In *International Conference on Learning Representations*, 2019. 39, 40, 46, 47, 49

[30] Dapeng Chen, Hongsheng Li, Tong Xiao, Shuai Yi, and Xiaogang Wang. Video person re-identification with competitive snippet-similarity aggregation and co-attentive snippet embedding. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2018. 64

[31] Ruihang Chu, Yifan Sun, Yadong Li, Zheng Liu, Chi Zhang, and Yichen Wei. Vehicle re-identification with viewpoint-aware metric learning. In *IEEE International Conference on Computer Vision*, 2019. 55, 64

[32] Djork-Arné Clevert, Thomas Unterthiner, and Sepp Hochreiter. Fast and accurate deep network learning by exponential linear units (elus). *International Conference on Learning Representations*, 2016. 13

[33] Patrizia Colangeli, Simona Iannetti, Angelo Cerella, Carla Ippoliti, and Alessio Di. Sistema nazionale di notifica delle malattie degli animali. *Veterinaria Italiana*, 2011. 77

[34] Yang Cong, Junsong Yuan, and Ji Liu. Sparse reconstruction cost for abnormal event detection. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*. IEEE, 2011. 20, 21, 22

[35] Annamaria Conte, Luca Candeloro, Carla Ippoliti, Federica Monaco, Fabrizio De Massis, Rossana Bruno, Daria Di Sabatino, Maria Luisa Danzetta, Abdennasser Benjelloun, Bouchra Belkadi, et al. Spatio-temporal identification of areas suitable for west nile disease in the mediterranean basin and central europe. *PloS one*, 2015. 77

[36] Michaël Defferrard, Xavier Bresson, and Pierre Vandergheynst. Convolutional neural networks on graphs with fast localized spectral filtering. In *Advances in Neural Information Processing Systems*, 2016. 4, 6, 15, 16

[37] Matthias Delange, Rahaf Aljundi, Marc Masana, Sarah Parisot, Xu Jia, Ales Leonardis, Greg Slabaugh, and Tinne Tuytelaars. A continual learning survey: Defying forgetting in classification tasks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021. 38, 40, 44, 48

[38] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*. IEEE, 2009. 69

[39] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2009. 83

[40] Aditya Deshpande, Jiajun Lu, Mao-Chuang Yeh, Min Jin Chong, and David Forsyth. Learning diverse image colorization. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2017. 72

[41] Carl Doersch, Abhinav Gupta, and Alexei A Efros. Unsupervised visual representation learning by context prediction. In *IEEE International Conference on Computer Vision*, 2015. 72

[42] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021. 84

[43] Matthias Drusch, Umberto Del Bello, Sébastien Carlier, Olivier Colin, Veronica Fernandez, Ferran Gascon, Bianca Hoersch, Claudia Isola, Paolo Laberinti, Philippe Martimort, et al. Sentinel-2: Esa's optical high-resolution mission for gmes operational services. *Remote sensing of Environment*, 2012. 69

[44] Yong Du, W. Wang, and L. Wang. Hierarchical recurrent neural network for skeleton based action recognition. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2015. 14

[45] Sebastian Farquhar and Yarin Gal. Towards Robust Evaluations of Continual Learning. In *ICML Workshops*, 2018. 40, 44

[46] Jan Feranec, Tomas Soukup, Gerard Hazeu, and Gabriel Jaffrain. *European landscape dynamics: CORINE land cover data*. CRC Press, 2016. 76

[47] Federico Filipponi. Exploitation of sentinel-2 time series to map burned areas at the national level: A case study on the 2017 italy wildfires. *Remote Sensing*, 2019. 69

[48] Yang Fu, Xiaoyang Wang, Yunchao Wei, and Thomas Huang. Sta: Spatial-temporal attention for large-scale video-based person re-identification. In *AAAI Conference on Artificial Intelligence*, 2019. 58, 62, 64

[49] Tommaso Furlanello, Zachary C Lipton, Michael Tschannen, Laurent Itti, and Anima Anandkumar. Born again neural networks. *International Conference on Machine Learning*, 2018. 56

[50] Mathieu Germain, Karol Gregor, Iain Murray, and Hugo Larochelle. Made: Masked autoencoder for distribution estimation. In *International Conference on Machine Learning*, 2015. 25

[51] Spyros Gidaris, Praveer Singh, and Nikos Komodakis. Unsupervised representation learning by predicting image rotations. In *International Conference on Learning Representations*, 2018. 72

[52] Tilmann Gneiting, Fadoua Balabdaoui, and Adrian E Raftery. Probabilistic forecasts, calibration and sharpness. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 2007. 51

[53] Xinqian Gu, Bingpeng Ma, Hong Chang, Shiguang Shan, and Xilin Chen. Temporal knowledge propagation for image-to-video person re-identification. In *IEEE International Conference on Computer Vision*, 2019. 53, 56, 57, 59, 61, 63, 64, 67

[54] Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. On calibration of modern neural networks. In *International Conference on Machine Learning*. JMLR. org, 2017. 51

[55] Renlong Hang, Qingshan Liu, Danfeng Hong, and Pedram Ghamisi. Cascaded recurrent neural networks for hyperspectral image classification. *IEEE Transactions on Geoscience and Remote Sensing*, 2019. 71

[56] Mahmudul Hasan, Jonghyun Choi, Jan Neumann, Amit K Roy-Chowdhury, and Larry S Davis. Learning temporal regularity in video sequences. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*. IEEE, 2016. 20, 21, 22, 32

[57] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *IEEE International Conference on Computer Vision*, 2015. 76

[58] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2016. 15, 34, 44, 57, 74

[59] Patrick Helber, Benjamin Bischke, Andreas Dengel, and Damian Borth. Eurosat: A novel dataset and deep learning benchmark for land use and land cover classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 2019. 76

[60] Alexander Hermans, Lucas Beyer, and Bastian Leibe. In defense of the triplet loss for person re-identification. In *IEEE International Conference on Computer Vision and Pattern Recognition Workshops*, 2017. 58, 61

[61] Ryota Hinami, Tao Mei, and Shin'ichi Satoh. Joint detection and recounting of abnormal events by learning deep generic knowledge. In *IEEE International Conference on Computer Vision*, 2017. 20, 22, 32

[62] Geoffrey Hinton, Li Deng, Dong Yu, George E Dahl, Abdel-rahman Mohamed, Navdeep Jaitly, Andrew Senior, Vincent Vanhoucke, Patrick Nguyen, Tara N Sainath, et al. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *IEEE Signal processing magazine*, 2012. 4

[63] Geoffrey Hinton, Oriol Vinyals, and Jeffrey Dean. Distilling the knowledge in a neural network. In *NeurIPS Deep Learning and Representation Learning Workshop*, 2015. 3, 39, 42, 47, 55, 56, 59

[64] Yen-Chang Hsu, Yen-Cheng Liu, Anita Ramasamy, and Zsolt Kira. Re-evaluating continual learning scenarios: A categorization and case for strong baselines. In *NeurIPS Continual learning Workshop*, 2018. 40, 44

[65] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2017. 62

[66] Minyoung Huh, Pulkit Agrawal, and Alexei A Efros. What makes imagenet good for transfer learning? *arXiv preprint arXiv:1608.08614*, 2016. 69

[67] David I Shuman, Sunil K. Narang, Pascal Frossard, Antonio Ortega, and Pierre Vandergheynst. The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains. *IEEE Signal Processing Magazine*, 2012. 5, 6

[68] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International Conference on Machine Learning*, 2015. 13

[69] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International Conference on Machine Learning*, 2015. 62

[70] Carla Ippoliti, Luca Candeloro, Marius Gilbert, Maria Goffredo, Giuseppe Mancini, Gabriele Curci, Serena Falasca, Susanna Tora, Alessio Di Lorenzo, Michela Quaglia, et al. Defining ecological regions in italy based on a multivariate clustering approach: A first step towards a targeted vector borne disease surveillance. *PloS one*, 2019. 69, 77

[71] Laurent Itti and Pierre Baldi. Bayesian surprise attracts human attention. *Vision research*, 2009. 20

[72] Stanisław Jastrzębski, Zachary Kenton, Devansh Arpit, Nicolas Ballas, Asja Fischer, Yoshua Bengio, and Amos Storkey. Three factors influencing minima in sgd. In *International Conference on Artificial Neural Networks*, 2018. 49, 50

[73] Neal Jean, Sherrie Wang, Anshul Samar, George Azzari, David Lobell, and Stefano Ermon. Tile2vec: Unsupervised representation learning for spatially distributed data. In *AAAI Conference on Artificial Intelligence*, 2019. 70

[74] Shunping Ji, Zhang Chi, Anjian Xu, Yun Shi, and Yulin Duan. 3d convolutional neural networks for crop classification with multi-temporal remote sensing images. *Remote Sensing*, 2018. 69

[75] Longlong Jing and Yingli Tian. Self-supervised visual feature learning with deep neural networks: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020. 72

[76] Thorsten Joachims. A probabilistic analysis of the rocchio algorithm with tfidf for text categorization. Technical report, Carnegie-mellon univ pittsburgh pa dept of computer science, CS7, 1996. 16

[77] Qiuhong Ke, Mohammed Bennamoun, Senjian An, Ferdous Ahmed Sohel, and Farid Boussaïd. A new representation of skeleton sequences for 3d action recognition. *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2017. 14

[78] Nitish Shirish Keskar, Dheevatsa Mudigere, Jorge Nocedal, Mikhail Smelyanskiy, and Ping Tak Peter Tang. On large-batch training for deep learning: Generalization gap and sharp minima. In *International Conference on Learning Representations*, 2017. 49, 50

[79] Sultan Daud Khan and Habib Ullah. A survey of advances in vision-based vehicle re-identification. *Computer Vision and Image Understanding*, 2019. 53

[80] Jaechul Kim and Kristen Grauman. Observe locally, infer globally: a space-time mrf for detecting abnormal activities with incremental updates. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*. IEEE, 2009. 22

[81] Tae Soo Kim and Austin Reiter. Interpretable 3d human action analysis with temporal convolutional networks. In *IEEE International Conference on Computer Vision and Pattern Recognition Workshops*. IEEE, 2017. 14

[82] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *International Conference on Learning Representations*, 2015. 13, 61

[83] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *International Conference on Learning Representations*, 2014. 30

[84] Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations*, 2017. 4, 6, 10, 16

[85] James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences*, 2017. 40, 44, 50

[86] Teuvo Kohonen. *Self-organization and associative memory*. Springer Science & Business Media, 2012. 21

[87] Alexander Kolesnikov, Xiaohua Zhai, and Lucas Beyer. Revisiting self-supervised visual representation learning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2019. 74

[88] Alex Krizhevsky et al. Learning multiple layers of features from tiny images. Technical report, Citeseer, 2009. 13, 44

[89] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*, 2012. 4

[90] Meelis Kull, Miquel Perello Nieto, Markus Kängsepp, Telmo Silva Filho, Hao Song, and Peter Flach. Beyond temperature scaling: Obtaining well-calibrated multi-class probabilities with dirichlet calibration. In *Advances in Neural Information Processing Systems*, 2019. 51

[91] Ajay Kumar. Computer-vision-based fabric defect detection: A survey. *IEEE Transactions on Industrial Electronics*, 2008. 20

[92] Junseok Kwon and Kyoung Mu Lee. A unified framework for event summarization and rare event detection from multiple views. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2015. 22

[93] Hugo Larochelle and Iain Murray. The neural autoregressive distribution estimator. In *International Conference on Artificial Intelligence and Statistics*, 2011. 24

[94] Gustav Larsson, Michael Maire, and Gregory Shakhnarovich. Learning representations for automatic colorization. In *Proceedings of the European Conference on Computer Vision*. Springer, 2016. 72, 74

[95] Gustav Larsson, Michael Maire, and Gregory Shakhnarovich. Colorization as a proxy task for visual understanding. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2017. 70, 72, 75

[96] Yann LeCun, Léon Bottou, Yoshua Bengio, Patrick Haffner, et al. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 1998. 44

[97] Chaolong Li, Zhen Cui, Wenming Zheng, Chunyan Xu, and Jian Yang. Spatio-temporal graph convolution for skeleton based action recognition. In *AAAI Conference on Artificial Intelligence*, 2018. 14

[98] Shuyuan Li, Jianguo Li, Weiyao Lin, and Hanlin Tang. Amur tiger re-identification in the wild. *arXiv preprint arXiv:1906.05586*, 2019. 61, 64

[99] Weixin Li, Vijay Mahadevan, and Nuno Vasconcelos. Anomaly detection and localization in crowded scenes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2014. 22

[100] Ying Li, Haokui Zhang, and Qiang Shen. Spectral–spatial classification of hyperspectral imagery with 3d convolutional neural network. *Remote Sensing*, 2017. 71

[101] Zhizhong Li and Derek Hoiem. Learning without forgetting. In *Proceedings of the European Conference on Computer Vision*, 2016. 59

[102] Zhizhong Li and Derek Hoiem. Learning without forgetting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017. 39, 40, 42, 46

[103] Shengcai Liao, Yang Hu, Xiangyu Zhu, and Stan Z Li. Person re-identification by local maximal occurrence representation and metric learning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2015. 64

[104] Zhouhan Lin, Yushi Chen, Xing Zhao, and Gang Wang. Spectral-spatial classification of hyperspectral image using autoencoders. In *International Conference on Information, Communications & Signal Processing*. IEEE, 2013. 72

[105] Zhouhan Lin, Roland Memisevic, and Kishore Reddy Konda. How far can we go without convolution: Improving fully-connected networks. In *IEEE International Conference on Computer Vision and Pattern Recognition Workshops*, 2016. 15

[106] Cen Liu, Rong Zhang, and Lijun Guo. Part-pose guided amur tiger re-identification. In *International Conference on Computer Vision Workshops*, 2019. 61, 64, 65

[107] Chih-Ting Liu, Chih-Wei Wu, Yu-Chiang Frank Wang, and Shao-Yi Chien. Spatially and temporally efficient non-local attention network for video-based person re-identification. In *British Machine Vision Conference*, 2019. 57, 58, 61, 64

[108] Jun Liu, Amir Shahroudy, Dong Xu, and Gang Wang. Spatio-temporal lstm with trust gates for 3d human action recognition. In *Proceedings of the European Conference on Computer Vision*, 2016. 14

[109] Ning Liu, Qijun Zhao, Nan Zhang, Xinhua Cheng, and Jianing Zhu. Pose-guided complementary features learning for amur tiger re-identification. In *International Conference on Computer Vision Workshops*, 2019. 64, 65

[110] Shichen Liu, Mingsheng Long, Jianmin Wang, and Michael I Jordan. Generalized zero-shot learning with deep calibration network. In *Advances in Neural Information Processing Systems*, 2018. 51

[111] Wen Liu, Weixin Luo, Dongze Lian, and Shenghua Gao. Future frame prediction for anomaly detection – a new baseline. *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2018. 20, 21, 22, 32

[112] Xiaobin Liu, Shiliang Zhang, Qingming Huang, and Wen Gao. Ram: a region-aware deep model for vehicle re-identification. In *IEEE International Conference on Multimedia and Expo*, 2018. 55, 60, 64

[113] Xinchen Liu, Wu Liu, Tao Mei, and Huadong Ma. A deep learning-based approach to progressive vehicle re-identification for urban surveillance. In *Proceedings of the European Conference on Computer Vision*, 2016. 54, 58, 61

[114] Xinchen Liu, Wu Liu, Tao Mei, and Huadong Ma. Provid: Progressive and multimodal vehicle reidentification for large-scale urban surveillance. *IEEE Transactions on Multimedia*, 2017. 55, 60, 64

[115] Xuan Liu, Xiaoguang Wang, and Stan Matwin. Improving the interpretability of deep neural networks with knowledge distillation. In *IEEE International Conference on Data Mining Workshops*. IEEE, 2018. 42

[116] Yu Liu, Yan Junjie, and Wanli Ouyang. Quality aware network for set to set recognition. In *IEEE International Conference on Computer Vision*, 2017. 57

[117] David Lopez-Paz and Marc'Aurelio Ranzato. Gradient episodic memory for continual learning. In *Advances in Neural Information Processing Systems*, 2017. 39, 40, 44, 46, 47

[118] Cewu Lu, Jianping Shi, and Jiaya Jia. Abnormal event detection at 150 fps in matlab. In *IEEE International Conference on Computer Vision*. IEEE, 2013. 22

[119] Pauline Luc, Natalia Neverova, Camille Couprie, Jakob Verbeek, and Yann LeCun. Predicting deeper into the future of semantic segmentation. In *IEEE International Conference on Computer Vision*, 2017. 24

[120] Hao Luo, Youzhi Gu, Xingyu Liao, Shenqi Lai, and Wei Jiang. Bag of tricks and a strong baseline for deep person re-identification. In *IEEE International Conference on Computer Vision and Pattern Recognition Workshops*, 2019. 57, 62

[121] Weixin Luo, Wen Liu, and Shenghua Gao. Remembering history with convolutional lstm for anomaly detection. In *IEEE International Conference on Multimedia and Expo*. IEEE, 2017. 32

[122] Weixin Luo, Wen Liu, and Shenghua Gao. A revisit of sparse coding based anomaly detection in stacked rnn framework. *IEEE International Conference on Computer Vision*, 2017. 20, 22, 32

[123] Xiaorui Ma, Hongyu Wang, and Jie Geng. Spectral–spatial classification of hyperspectral image based on deep auto-encoder. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 2016. 72

[124] Vijay Mahadevan, Weixin Li, Viral Bhalodia, and Nuno Vasconcelos. Anomaly detection in crowded scenes. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*. IEEE, 2010. 22, 32

[125] Konstantinos Makantasis, Konstantinos Karantzalos, Anastasios Doulamis, and Nikolaos Doulamis. Deep supervised learning for hyperspectral data classification through convolutional neural networks. In *IEEE International Geoscience and Remote Sensing Symposium*. IEEE, 2015. 71, 72

[126] Arun Mallya and Svetlana Lazebnik. Packnet: Adding multiple tasks to a single network by iterative pruning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2018. 40

[127] Dimitrios Marmanis, Mihai Datcu, Thomas Esch, and Uwe Stilla. Deep learning earth observation classification using imagenet pretrained networks. *IEEE Geoscience and Remote Sensing Letters*, 2015. 71

[128] Giuseppe Masi, Davide Cozzolino, Luisa Verdoliva, and Giuseppe Scarpa. Pansharpening by convolutional neural networks. *Remote Sensing*, 2016. 69

[129] Neeraj Matiyali and Gaurav Sharma. Video person re-identification using learned clip similarity aggregation. In *Proceedings of the IEEE Winter conference on Applications of Computer Vision*, 2020. 61, 64

[130] Michael McCloskey and Neal J Cohen. Catastrophic interference in connectionist networks: The sequential learning problem. In *Psychology of learning and motivation*. Elsevier, 1989. 3, 38

[131] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *International Conference on Learning Representations*, 2013. 16

[132] Federico Monti, Davide Boscaini, Jonathan Masci, Emanuele Rodola, Jan Svoboda, and Michael M Bronstein. Geometric deep learning on graphs and manifolds using mixture model cnns. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2017. 6

[133] Mahdi Pakdaman Naeini, Gregory Cooper, and Milos Hauskrecht. Obtaining well calibrated probabilities using bayesian binning. In *AAAI Conference on Artificial Intelligence*, 2015. 51

[134] Maxim Neumann, Andre Susano Pinto, Xiaohua Zhai, and Neil Houlsby. In-domain representation learning for remote sensing. *arXiv preprint arXiv:1911.06721*, 2019. 70, 71, 77, 79

[135] Thuy-Binh Nguyen, Thi-Lan Le, Dinh-Duc Nguyen, and Dinh-Tan Pham. A reliable image-to-video person re-identification based on feature fusion. In *Asian conference on intelligent information and database systems*, 2018. 53

[136] Mathias Niepert, Mohamed Ahmed, and Konstantin Kutzkov. Learning convolutional neural networks for graphs. In *International Conference on Machine Learning*. PMLR, 2016. 7

[137] Keiller Nogueira, Otávio AB Penatti, and Jefersson A Dos Santos. Towards better exploiting convolutional neural networks for remote sensing scene classification. *Pattern Recognition*, 2017. 71

[138] Mehdi Noroozi and Paolo Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. In *Proceedings of the European Conference on Computer Vision*. Springer, 2016. 72

[139] Andrea Palazzi, Davide Abati, Simone Calderara, Francesco Solera, and Rita Cucchiara. Predicting the driver's focus of attention: the dr(eye)ve project. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2018. 36

[140] Jongchan Park, Sanghyun Woo, Joon-Young Lee, and In So Kweon. BAM: bottleneck attention module. In *British Machine Vision Conference*, 2018. 63

[141] Deepak Pathak, Pulkit Agrawal, Alexei A Efros, and Trevor Darrell. Curiosity-driven exploration by self-supervised prediction. In *International Conference on Machine Learning*, 2017. 20

[142] Angelo Porrello, Davide Abati, Simone Calderara, and Rita Cucchiara. Classifying signals on irregular domains via convolutional cluster pooling. In *International Conference on Artificial Intelligence and Statistics*. PMLR, 2019. 2, 85

[143] Angelo Porrello, Luca Bergamini, and Simone Calderara. Robust re-identification by multiple views knowledge distillation. In *Proceedings of the European Conference on Computer Vision*. Springer, 2020. 3, 86

[144] Angelo Porrello, Stefano Vincenzi, Pietro Buzzega, Simone Calderara, Annamaria Conte, Carla Ippoliti, Luca Candeloro, Alessio Di Lorenzo, and Andrea Capobianco Dondona. Spotting insects from satellites: modeling the presence of culicoides imicola through deep cnns. In *International Conference on Signal-Image Technology & Internet-Based Systems*. IEEE, 2019. 77, 78

[145] Geesara Prathap and Ilya Afanasyev. Deep learning approach for building detection in satellite multispectral imagery. In *International Conference on Intelligent Systems*. IEEE, 2018. 78

[146] Jingjing Qian, Wei Jiang, Hao Luo, and Hongyan Yu. Stripe-based and attribute-aware network: A two-branch deep model for vehicle re-identification. *Measurement Science and Technology*, 2020. 63, 64

[147] Roger Ratcliff. Connectionist models of recognition memory: constraints imposed by learning and forgetting functions. *Psychological review*, 1990. 39, 40

[148] Mahdyar Ravanbakhsh, Enver Sangineto, Moin Nabi, and Nicu Sebe. Training adversarial discriminators for cross-channel abnormal event detection in crowds. *Proceedings of the IEEE Winter conference on Applications of Computer Vision*, 2019. 23

[149] Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H Lampert. icarl: Incremental classifier and representation learning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2017. 39, 40, 42, 44, 45, 46

[150] Flavia Riccardo, Federica Monaco, Antonino Bella, Giovanni Savini, Francesca Russo, Roberto Cagarelli, Michele Dottori, Caterina Rizzo, Giulietta Venturi, Marco Di Luca, et al. An early start of west nile virus seasonal transmission: the added value of one heath surveillance in detecting early circulation and triggering timely response in italy, june to july 2018. *Eurosurveillance*, 2018. 77

[151] Matthew Riemer, Ignacio Cases, Robert Ajemian, Miao Liu, Irina Rish, Yuhai Tu, and Gerald Tesauro. Learning to learn without forgetting by maximizing transfer and minimizing interference. In *International Conference on Learning Representations*, 2019. 39, 40, 41, 44, 46, 49

[152] Ergys Ristani, Francesco Solera, Roger Zou, Rita Cucchiara, and Carlo Tomasi. Performance measures and a data set for multi-target, multi-camera tracking. In *Proceedings of the European Conference on Computer Vision*, 2016. 58, 61

[153] Ergys Ristani and Carlo Tomasi. Features for multi-target multi-camera tracking and re-identification. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2018. 61

[154] Anthony Robins. Catastrophic forgetting, rehearsal and pseudorehearsal. *Connection Science*, 1995. 39

[155] David Rolnick, Priya L Donti, Lynn H Kaack, Kelly Kochanski, Alexandre Lacoste, Kris Sankaran, Andrew Slavin Ross, Nikola Milojevic-Dupont, Natasha Jaques, Anna Waldman-Brown, et al. Tackling climate change with machine learning. *arXiv preprint arXiv:1906.05433*, 2019. 69

[156] Adriana Romero, Nicolas Ballas, Samira Ebrahimi Kahou, Antoine Chassang, Carlo Gatta, and Yoshua Bengio. Fitnets: Hints for thin deep nets. *International Conference on Learning Representations*, 2015. 55

[157] Andrei A Rusu, Neil C Rabinowitz, Guillaume Desjardins, Hubert Soyer, James Kirkpatrick, Koray Kavukcuoglu, Razvan Pascanu, and Raia Hadsell. Progressive neural networks. *arXiv preprint arXiv:1606.04671*, 2016. 40, 46

[158] Inderjit S Dhillon, Yuqiang Guan, and Brian Kulis. Weighted graph cuts without eigenvectors a multilevel approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2007. 6, 17

[159] Mohammad Sabokrou, Mohsen Fayyaz, Mahmood Fathy, Zahra Moayed, and Reinhard Klette. Deep-anomaly: Fully convolutional neural network for fast anomaly detection in crowded scenes. *Computer Vision and Image Understanding*, 2018. 22

[160] Mohammad Sabokrou, Mohammad Khalooei, Mahmood Fathy, and Ehsan Adeli. Adversarially learned one-class classifier for novelty detection. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2018. 20, 21, 22

[161] Wojciech Samek, Thomas Wiegand, and Klaus-Robert Müller. Explainable artificial intelligence: Understanding, visualizing and interpreting deep learning models. *arXiv preprint arXiv:1708.08296*, 2017. 81

[162] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2018. 59, 62

[163] Thomas Schlegl, Philipp Seeböck, Sebastian M Waldstein, Ursula Schmidt-Erfurth, and Georg Langs. Unsupervised anomaly detection with generative adversarial networks to guide marker discovery. In *International Conference on Information Processing in Medical Imaging*. Springer, 2017. 23, 30

[164] Stefan Schneider, Graham W Taylor, Stefan Linquist, and Stefan C Kremer. Past, present and future approaches using computer vision for animal re-identification from camera trap data. *Methods in Ecology and Evolution*, 2019. 53

[165] Bernhard Schölkopf, Robert C Williamson, Alex J Smola, John Shawe-Taylor, and John C Platt. Support vector method for novelty detection. In *Advances in Neural Information Processing Systems*, 2000. 29

[166] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2015. 53

[167] Guy JP Schumann, G Robert Brakenridge, Albert J Kettner, Rashid Kashif, and Emily Niebuhr. Assisting flood disaster response with earth observation data and products: a critical assessment. *Remote Sensing*, 2018. 69

[168] Jonathan Schwarz, Wojciech Czarnecki, Jelena Luketina, Agnieszka Grabska-Barwinska, Yee Whye Teh, Razvan Pascanu, and Raia Hadsell. Progress & compress: A scalable framework for continual learning. In *International Conference on Machine Learning*, 2018. 40, 46

[169] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the*

IEEE conference on Computer Vision and Pattern Recognition, 2017. 67, 68, 81

[170] Joan Serra, Didac Suris, Marius Miron, and Alexandros Karatzoglou. Overcoming catastrophic forgetting with hard attention to the task. In *International Conference on Machine Learning*. PMLR, 2018. 40

[171] Amir Shahroudy, Jun Liu, Tian-Tsong Ng, and Gang Wang. Ntu rgb+d: A large scale dataset for 3d human activity analysis. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2016. 13, 14

[172] Jamie Sherrah. Fully convolutional networks for dense semantic labelling of high-resolution aerial imagery. *arXiv preprint arXiv:1606.02585*, 2016. 69

[173] Jianlou Si, Honggang Zhang, Chun-Guang Li, Jason Kuen, Xiangfei Kong, Alex C Kot, and Gang Wang. Dual attention matching network for context-aware feature sequence based person re-identification. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2018. 61, 64

[174] Kihyuk Sohn. Improved deep metric learning with multi-class n-pair loss objective. In *Neural Information Processing Systems*, 2016. 53

[175] Stanford. Tiny ImageNet Challenge (CS231n), 2015. `http://tiny-imagenet.herokuapp.com/`. 44

[176] Felipe Petroski Such, Shagan Sah, Miguel Domínguez, Suhas Pillai, Chao Zhang, Andrew Michael, Nathan D. Cahill, and Raymond W. Ptucha. Robust spatial filtering with graph convolutional neural networks. *IEEE Journal of Selected Topics in Signal Processing*, 2017. 6, 15

[177] Gencer Sumbul, Marcela Charfuelan, Begüm Demir, and Volker Markl. Bigearthnet: A large-scale benchmark archive for remote sensing image understanding. In *IEEE International Geoscience and Remote Sensing Symposium*. IEEE, 2019. 70, 76, 81

[178] Gencer Sumbul, Jian Kang, Tristan Kreuziger, Filipe Marcelino, Hugo Costa, Pedro Benevides, Mario Caetano, and Begüm Demir. Bigearthnet deep learning models with a new class-nomenclature for remote sensing image understanding. *arXiv preprint arXiv:2001.06372*, 2020. 76, 77

[179] Zheng Tang, Milind Naphade, Stan Birchfield, Jonathan Tremblay, William Hodge, Ratnesh Kumar, Shuo Wang, and Xiaodong Yang. Pamtri: Pose-aware multi-task learning for vehicle re-identification using highly randomized synthetic data. In *IEEE International Conference on Computer Vision*, 2019. 64

[180] Graham W Taylor, Rob Fergus, Yann LeCun, and Christoph Bregler. Convolutional learning of spatio-temporal features. In *Proceedings of the European Conference on Computer Vision*. Springer, 2010. 4

[181] Lucas Theis, Aäron van den Oord, and Matthias Bethge. A note on the evaluation of generative models. *International Conference on Learning Representations*, 2016. 31

[182] Maoqing Tian, Shuai Yi, Hongsheng Li, Shihua Li, Xuesen Zhang, Jianping Shi, Junjie Yan, and Xiaogang Wang. Eliminating background-bias for robust person re-identification. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2018. 65

[183] Jakub M Tomczak and Max Welling. Vae with a vamp prior. *International Conference on Artificial Intelligence and Statistics*, 2018. 22

[184] Annelise Tran, Bertrand Sudre, Shlomit Paz, Massimiliano Rossi, Annie Desbrosse, Véronique Chevalier, and Jan C Semenza. Environmental predictors of west nile fever risk in europe. *International journal of health geographics*, 2014. 77

[185] Myron Tribus. *Thermostatics and thermodynamics: an introduction to energy, information and states of matter, with engineering applications*. van Nostrand, CS7, 1961. 20, 23

[186] Frederick Tung and Greg Mori. Similarity-preserving knowledge distillation. In *IEEE International Conference on Computer Vision*, 2019. 55, 56

[187] Benigno Uria, Iain Murray, and Hugo Larochelle. Rnade: The real-valued neural autoregressive density-estimator. In *Advances in Neural Information Processing Systems*, 2013. 24

[188] Benigno Uria, Iain Murray, and Hugo Larochelle. A deep and tractable density estimator. In *International Conference on Machine Learning*, 2014. 25

[189] Evgeniya Ustinova and Victor Lempitsky. Learning deep embeddings with histogram loss. In *Advances in Neural Information Processing Systems*, 2016. 53

[190] Gido M van de Ven and Andreas S Tolias. Three continual learning scenarios. *NeurIPS Continual Learning Workshop*, 2018. 40, 44

[191] A. van den Oord, N. Kalchbrenner, L. Espeholt, O. Vinyals, and A. Graves. Conditional image generation with pixelcnn decoders. In *Advances in Neural Information Processing Systems*, 2016. 25, 30

[192] A. van den Oord, Nal Kalchbrenner, and Koray Kavukcuoglu. Pixel recurrent neural networks. *International Conference on Machine Learning*, 2016. 25

[193] Aaron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, and Koray Kavukcuoglu. WaveNet: A Generative Model for Raw Audio. In *Proc. 9th ISCA Workshop on Speech Synthesis Workshop*, 2016. 24

[194] Petar Velikovic, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph attention networks. *International Conference on Learning Representations*, 2018. 17

[195] Raviteja Vemulapalli, Felipe Arrate, and Rama Chellappa. Human action recognition by representing 3d skeletons as points in a lie group. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2014. 14

[196] Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. Extracting and composing robust features with denoising autoencoders. In *International Conference on Machine Learning*, 2008. 72

[197] Stefano Vincenzi, Angelo Porrello, Pietro Buzzega, Marco Cipriano, Pietro Fronte, Roberto Cuccu, Carla Ippoliti, Annamaria Conte, and Simone Calderara. The color out of space: learning self-supervised representations for earth observation imagery. In *International Conference on Pattern Recognition*. IEEE, 2021. 3, 86

[198] Jeffrey S Vitter. Random sampling with a reservoir. *ACM Transactions on Mathematical Software (TOMS)*, 1985. 41

[199] Ulrike Von Luxburg. A tutorial on spectral clustering. *Statistics and computing*, 2007. 8

[200] Guangcong Wang, Jianhuang Lai, and Xiaohua Xie. P2snet: can an image match a video for person re-identification in an end-to-end way? *IEEE Transactions on Circuits and Systems for Video Technology*, 2017. 55, 64

[201] Jian Wang, Feng Zhou, Shilei Wen, Xiao Liu, and Yuanqing Lin. Deep metric learning with angular loss. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2017. 53

[202] Zhongdao Wang, Luming Tang, Xihui Liu, Zhuliang Yao, Shuai Yi, Jing Shao, Junjie Yan, Shengjin Wang, Hongsheng Li, and Xiaogang Wang. Orientation invariant feature embedding and spatial temporal regularization for vehicle re-identification. In *IEEE International Conference on Computer Vision*, 2017. 60

[203] Yu Wu, Yutian Lin, Xuanyi Dong, Yan Yan, Wanli Ouyang, and Yi Yang. Exploit the unknown gradually: One-shot video-based person re-identification by stepwise learning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2018. 54, 61

[204] Yue Wu, Yinpeng Chen, Lijuan Wang, Yuancheng Ye, Zicheng Liu, Yandong Guo, and Yun Fu. Large scale incremental learning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2019. 44, 45

[205] Zhongwei Xie, Lin Li, Xian Zhong, Luo Zhong, and Jianwen Xiang. Image-to-video person re-identification with cross-modal embeddings. *Pattern Recognition Letters*, 2019. 53

[206] Chenglin Yang, Lingxi Xie, Siyuan Qiao, and Alan Yuille. Knowledge distillation in generations: More tolerant teachers educate better students. *arXiv preprint arXiv:1805.05551*, 2018. 56

[207] Yi Yang and Shawn Newsam. Bag-of-visual-words and spatial extensions for land-use classification. In *Proceedings of the 18th SIGSPATIAL international conference on advances in geographic information systems*, 2010. 76

[208] Jiwen Yu, Haibo Su, Junnan Liu, Zhizheng Yang, Zhouyangzi Zhang, Yixin Zhu, Lu Yang, and Bingliang Jiao. A strong baseline for tiger re-id and its bag of tricks. In *International Conference on Computer Vision Workshops*, 2019. 64, 65

[209] Xingrui Yu, Xiaomin Wu, Chunbo Luo, and Peng Ren. Deep learning in remote sensing scene classification: a data augmentation enhanced convolutional neural network framework. *GIScience & Remote Sensing*, 2017. 71

[210] Sergey Zagoruyko and Nikos Komodakis. Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer. In *International Conference on Learning Representations*, 2017. 55

[211] Matthew Zeiler and Robert Fergus. Stochastic pooling for regularization of deep convolutional neural networks. In *International Conference on Learning Representations*, 2013. 15

[212] Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *Proceedings of the European Conference on Computer Vision*. Springer, 2014. 33

[213] Friedemann Zenke, Ben Poole, and Surya Ganguli. Continual learning through synaptic intelligence. In *International Conference on Machine Learning*, 2017. 40, 44, 45, 46

[214] Dongyu Zhang, Wenxi Wu, Hui Cheng, Ruimao Zhang, Zhenjiang Dong, and Zhaoquan Cai. Image-to-video person re-identification with temporally memorized similarity learning. *IEEE Transactions on Circuits and Systems for Video Technology*, 2017. 53, 55, 57, 64

[215] Linfeng Zhang, Jiebo Song, Anni Gao, Jingwei Chen, Chenglong Bao, and Kaisheng Ma. Be your own teacher: Improve the performance of convolutional neural networks via self distillation. In *IEEE International Conference on Computer Vision*, 2019. 50

[216] Richard Zhang, Phillip Isola, and Alexei A Efros. Colorful image colorization. In *Proceedings of the European Conference on Computer Vision*. Springer, 2016. 74, 75

[217] Richard Zhang, Phillip Isola, and Alexei A Efros. Split-brain autoencoders: Unsupervised learning by cross-channel prediction. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2017. 70, 73, 75

[218] Tong Zhang, Wenming Zheng, Zhen Cui, and Yang Li. Tensor graph convolutional neural network. *arXiv preprint arXiv:1803.10071*, 2018. 14

[219] Ying Zhang, Tao Xiang, Timothy M Hospedales, and Huchuan Lu. Deep mutual learning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2018. 50

[220] Bin Zhao, Li Fei-Fei, and Eric P Xing. Online detection of unusual events in videos via dynamic sparse coding. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*. IEEE, 2011. 22

[221] Liang Zheng, Zhi Bie, Yifan Sun, Jingdong Wang, Chi Su, Shengjin Wang, and Qi Tian. Mars: A video benchmark for large-scale person re-identification. In *Proceedings of the European Conference on Computer Vision*, 2016. 54, 58, 61

[222] Liang Zheng, Yi Yang, and Alexander G Hauptmann. Person re-identification: Past, present and future. *arXiv preprint arXiv:1610.02984*, 2016. 53, 61

[223] Zhun Zhong, Liang Zheng, Donglin Cao, and Shaozi Li. Re-ranking person re-identification with k-reciprocal encoding. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2017. 65

[224] Weixun Zhou, Shawn Newsam, Congmin Li, and Zhenfeng Shao. Patternnet: A benchmark dataset for performance evaluation of remote sensing image retrieval. *ISPRS journal of photogrammetry and remote sensing*, 2018. 76

[225] Yi Zhou, Li Liu, and Ling Shao. Vehicle re-identification by deep hidden multi-view inference. *IEEE Transactions on Image Processing*, 2018. 60

[226] Yi Zhou and Ling Shao. Aware attentive multi-view inference for vehicle re-identification. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2018. 55, 60

[227] Bo Zong, Qi Song, Martin Renqiang Min, Wei Cheng, Cristian Lumezanu, Daeki Cho, and Haifeng Chen. Deep autoencoding gaussian mixture model for unsupervised anomaly detection. In *International Conference on Learning Representations*, 2018. 20, 22