Few-Shot Fast-Adaptive Anomaly Detection

Ze Wang †, Yipin Zhou †, Rui Wang †, Tsung-Yu Lin †, Ashish Shah †, and Ser-Nam Lim ‡
†Purdue University ‡Meta AI

Abstract

The ability to detect anomaly has long been recognized as an inherent human ability, yet to date, practical AI solutions to mimic such capability have been lacking. This lack of progress can be attributed to several factors. To begin with, the distribution of “abnormalities” is intractable. Anything outside of a given normal population is by definition an anomaly. This explains why a large volume of work in this area has been dedicated to modeling the normal distribution of a given task followed by detecting deviations from it. This direction is however unsatisfying as it would require modeling the normal distribution of every task that comes along, which includes tedious data collection. In this paper, we report our work aiming to handle these issues. To deal with the intractability of abnormal distribution, we leverage Energy Based Model (EBM). EBMs learn to associate low energies to correct values and higher energies to incorrect values. At its core, the EBM employs Langevin Dynamics (LD) in generating these incorrect samples based on an iterative optimization procedure, alleviating the intractable problem of modeling the world of anomalies. Then, in order to avoid training an anomaly detector for every task, we utilize an adaptive sparse coding layer. Our intention is to design a plug and play feature that can be used to quickly update what is normal during inference time. Lastly, to avoid tedious data collection, this mentioned update of the sparse coding layer needs to be achievable with just a few shots. Here, we employ a meta learning scheme that simulates such a few shot setting during training. We support our findings with strong empirical evidence.

1 Introduction

Anomaly detection is an important area of study in the field of artificial intelligence. It has found utility in computer vision applications such as industrial inspection [6] and video surveillance [28, 61, 59], in the context of abuse prevention such as misinformation, fraud and network intrusion detection [60, 8, 35], and others such as system health monitoring and fault detection [4, 42]. In this paper, we propose an approach for detecting anomaly in images, where we have carefully designed steps to handle some of the bigger issues that have prevented the deployment of image anomaly detection in the real-world.

Image anomaly detection can generally be defined as the identification of abnormalities in a given image. An exact definition of abnormality in this case is elusive because abnormality can be derived from any unknown distribution outside of a normal population. Many studies have hence focused on modeling the normal population instead of learning irregularities, where the goal is to capture the shared concept among all of the normal data as one or several reference models. This process usually requires investing significant efforts in curating a large set of normal samples for each task, after which anomaly is detected as deviations from the reference model(s) [1, 58]. Recent work from [50] provides algorithms that utilize only a few normal samples to train models from scratch. However, the models still have to be provisioned for each new task, which requires considerable human efforts

*Work done as an intern at Meta AI. Contact: zewang@purdue.edu

and expertise, and thus lack the fast deployment criterion that is often time critical for real-world applications. In view of these challenges, our goals for this work are threefold. We are interested in designing an anomaly detection system that is capable of: (G1) modeling the normal population while at the same time has a principled approach towards modeling the abnormalities; (G2) quickly adapting to a new task at inference time; and (G3) requiring only a few normal shots to update itself to the new task at hand.

For (G1), we introduce the class of Energy Based Model (EBM), which is an important family of generative models \[ \text{\cite{62,17,57}} \]. EBMs have been shown to demonstrate superior capability on modeling data density and localizing anomaly \[ \text{\cite{20}} \]. For our purpose, the EBM we adopted learns to assign low energy to normal samples but high energy to abnormal samples. More importantly, the abnormal samples are generated with a procedure known as Langevin Dynamics (LD) \[ \text{\cite{54}} \], which, in its original form, starts with a noise image and gradually samples from the distribution along the direction of lower energy. This lends itself gracefully to utilizing the generated intermediate samples as negative/abnormal. The LD procedure is then coupled with maximum likelihood loss \[ \text{\cite{24}} \] that aims to maximize the energy differences between the normal and abnormal samples.

To achieve (G2), we propose an adaptive sparse coding layer that is attached to the deep feature extractor in the EBM as Figure 1 shows. The extracted deep feature is forwarded to the sparse coding layer, where the dictionary is constructed with the features of a few normal samples of the given task. In essence, the input representation has been decomposed into a linear combination of normal features with the sparsity constraint imposed. The final energy score is measured by the distance between the original and the reconstructed features (after the sparse coding layer). Under this scheme, the dictionary for a particular task is not obtained by learning, but instead is constructed by the feature representations of a few normal samples during inference. As a result, this simple “plug-and-play” trick allows the model to be adapted to novel tasks promptly without re-training. Further, we expect that the dictionary, which is formed by normal features, will not be able to explain the abnormal samples well, causing relatively high reconstruction error that lends itself for subsequent detection. As a bonus, a backward pass of energy score minimization can be used for localizing abnormal regions. We show that using gradient to localize anomalies yields superior robustness.

Towards (G3), we utilize meta learning \[ \text{\cite{52,18}} \] to simulate the scenario of being given a new task with a few normal shots to update the dictionary, followed by training the EBM. This is accomplished by episodic training, where in each episode the model is adapted to a held back task that is given a few normal samples. To accelerate the EBM training, we introduce “learning from inpainting”, a simple yet effective strategy for synthesizing hard abnormal samples quicker by starting the LD procedure with a synthesized image that is simply a normal sample with a noise patch injected as opposed to a noise image that is traditionally what is used.

We show the proposed few-shot fast-adaptive anomaly detection and localization framework is able to efficiently adapt to a novel task (e.g., a new object category or scenes from a new camera) with a few normal samples without training on both industrial inspection and video surveillance. Compared with previous methods that adapts to new task through either from scratch training in few shots \[ \text{\cite{50,55}} \] or few-shot with few steps of gradient descent \[ \text{\cite{32}} \], the proposed framework is the first that performs task adaptation with a single forward pass and without any gradient descent. Despite the fast adaptation, we provide both qualitative and quantitative results to demonstrate that our method outperforms other adaptive frameworks and is comparable to methods that rely on large amount of normal samples.

## 2 Backgrounds

In this section, we briefly introduce two key ingredients of the proposed method: EBMs and sparse coding.

**Energy-based Model.** In EBMs, the goal is to learn an energy function \( E_\theta(x) : \mathbb{R}^d \to \mathbb{R} \) which parametrizes the data density \( p_\theta(x) \) as:

\[
p_\theta(x) = \frac{\exp(-E_\theta(x))}{\int_x \exp(-E_\theta(x))},
\]

where \( \theta \) is the parameter of the energy function and \( Z_\theta = \int_x \exp(-E_\theta(x)) \) is the partition function. Approximating the true data distribution \( p_{\text{data}}(x) \) is equivalent to minimizing the expected negative
log-likelihood function over the data distribution, defined by the loss function:
\[
\mathcal{L}_{\text{ML}} = \mathbb{E}_{x \sim p_{\text{data}}(x)}[-\log p_{\theta}(x)] = \mathbb{E}_{x \sim p_{\text{data}}(x)}[E_{\theta}(x) + \log Z_{\theta}].
\] (2)

As the computation of \( \mathcal{L}_{\text{ML}} \) involves an intractable term \( Z_{\theta} \), the common practice is to represent the gradient of \( \mathcal{L}_{\text{ML}} \) as:
\[
\nabla_{\theta} \mathcal{L}_{\text{ML}} = \mathbb{E}_{x \sim p_{\text{data}}(x)}[\nabla_{\theta} E_{\theta}(x^+)] - \mathbb{E}_{x \sim p_{\text{data}}(x)}[\nabla_{\theta} E_{\theta}(x^-)].
\] (3)

This objective decreases the energy of positive data samples \( x^+ \) from the true distribution (normal samples in our use case) and increases the energy of negative samples \( x^- \) from the model \( p_{\theta} \) (synthesized abnormal samples). In practice, the synthesized negative samples are achieved through Langevin dynamics \([24]\), which a \( J \)-steps sampling along the direction of energy minimization is given by:
\[
\hat{x}^j = \hat{x}^{j-1} - \beta \nabla_x \mathbb{E}_{\theta}(\hat{x}^{j-1}) + \omega^k, \quad \omega^k \sim \mathcal{N}(0, \beta I), \quad j = \{1, \ldots, J\}
\] (4)

where \( \beta \) is the step size, and the initialization \( x^0 \) is sampled from a predefined prior distribution. The synthesizing ability of EBMs enables generating abnormal samples to help in learning a more accurate data density, and is often touted as the one of the advantages of using an EBM.

**Sparse coding.** Approximating a signal \( z \in \mathbb{R}^d \) with the sparse linear combination over a dictionary \( D \in \mathbb{R}^{d \times k} \) can be expressed as:
\[
\min_{\alpha} \frac{1}{2}||z - D \alpha||^2 + \lambda ||\alpha||_1,
\] (5)

where \( \alpha \) is the sparse coefficients, with its sparsity (\( l_1 \) norm) and \( \lambda \) is the weight of the sparsity constraint. \( D \alpha \) is a sparse approximation to the original signal \( z \). In practice, finding the dictionary atoms and the sparse coefficients is usually formulated as an optimization problem.

In this paper, we adopt Iterative Soft Thresholding Continuation (ISTC) \([25]\) to convert this optimization problem into linear operations with a non-linear shrinkage function, which allows sparse coding to be seamlessly integrated into the deep neural networks. To compute a sparse coefficient \( \alpha \), ISTC performs iterations of gradient steps on reconstruction \( ||z - D \alpha||^2 \) and a proximal projection step to increase coefficient sparsity.

Formally, initializing the coefficients at the first step \( \alpha_0 \) with all zeros, each step of ISTC refines the sparse code with descending values of \( \lambda \) from \( \lambda_{\text{max}} \) to \( \lambda^*_s \): each step of ISTC is expressed as:
\[
\alpha_{n+1} = \sigma(\alpha_n + D^T(z - D \alpha_n), \lambda_n), \quad \text{with} \quad \lambda_n = \lambda_{\text{max}} \frac{\lambda_{\text{max}} - n/N}{\lambda^*_s},
\] (6)

where \( \sigma(\cdot, \cdot) \) here is a shrinkage function that truncates small values (lower than \( \lambda \)) of the coefficients to 0 to enforce sparsity, and can be easily implemented by a customized ReLU activation function:
\[
\sigma(z, \lambda) = \text{sgn}(z)(\max(|z| - \lambda, 0)) = \text{sgn}(z)\text{ReLU}(|z| - \lambda).
\] (7)

### 3 Proposed Method

In this section, we describe the proposed fast adaptive anomaly detection framework in details. In Section 3.1, we introduce the adaptive EBM which consists of a deep feature extractor followed by an adaptive sparse coding layer. From there, we further show that utilizing larger receptive field in the sparse coding could improve training robustness (Section 3.1.1), and applying smoothed shrinkage functions could help speed up convergence (Section 3.1.2). In Section 3.2, we describe the episodic training regime on various anomaly detection tasks that mimics few-shot adaptation in the meta-testing stage while learning common knowledge across tasks. Instead of synthesizing negative samples (anomaly) directly from noise, we introduce a simple but effective “learning from inpainting” operation to accelerate the training in Section 3.3. Finally, we summarize training steps of the proposed method in Algorithm 3. and inference steps in Algorithm 4.
3.1 Adaptive Energy-based Model

An EBM is a form of generative model and it is widely used for modeling data density and sampling. While there has been recent work applying EBM to anomaly detection, it still requires re-training for each new task. To efficiently adapt the EBM to novel tasks, we introduce an adaptive sparse coding layer which is conditioned on the dictionary constructed by the features of normal samples. Specifically, as illustrated in Fig 1(a), given an input image, \( x \in \mathbb{R}^{d \times h \times w} \), we first obtain the corresponding feature vector \( z \in \mathbb{R}^{d' \times h' \times w'} \) from the deep feature extractor \( \Psi \) with parameters \( \theta \), so that \( z = \Psi(x; \theta) \). All feature vectors along spatial axes of \( z \) are then sparsely decomposed through the sparse coding layer over a task-specific dictionary \( D \in \mathbb{R}^{d' \times K h' w'} \), which contains the features of \( K \) normal samples of the current task as shown in the Fig 1(a). Each feature vector of the normal sample features is then directly used as an atom in the task dictionary. The decomposed coefficients are \( \alpha = S(z; D) \), where \( \alpha \in \mathbb{R}^{K h' w' \times h' \times w'} \) and \( S \) denotes the iterative sparse decomposition process of \( \Psi \). By multiplying the coefficient \( \alpha \) with the dictionary \( D \), we obtain the reconstructed features \( z' = D \alpha \). The sparsity regularization to \( \alpha \) is important, as it encourages input features to be reconstructed by simple combinations of dictionary atoms (normal features), so that it would be difficult for features of abnormal samples to be well-approximated, therefore producing higher reconstruction errors that make it conducive for detecting anomalies. From here, the final energy score is formulated as the mean squared error (MSE) between the original and the reconstructed features:

\[
E_\theta(x; D) = \text{MSE}(z, z') = \|\Psi(x; \theta) - DS(\Psi(x; \theta); D)\|^2.
\]

In effect, Eqn. 8 depicts a conditional EBM, which is conditioned on the task-specific \( D \) formed by normal features. With the energy score, we can obtain pixel-wise anomaly localization maps through \( \nabla_x E_\theta(x; D) \), i.e., the gradients of pixels along the direction of minimization. High gradient magnitudes indicate regions that cannot be well explained by the dictionary \( D \). Modifications to these regions can potentially remove the anomaly and reduce the energy as in Eqn. 8. In Section 4.1 and Appendix Section B.5, we show that using the gradient (as a natural ingredient of EBMs with LD) is more robust compared with auto-encoder and reconstruction based methods to generalize well to unseen tasks (Appendix Figure C). In the following sections, we will discuss how to make the training of this adaptive structure more robust.

3.1.1 Sparse Coding with Receptive Field.

As discussed in Section 3.1, the input feature \( z \) is represented as \( h' \times w' \) of \( d \)-dim feature vectors and they are treated independently while passing through the sparse coding layer. The region of the input image that affects one feature vector is determined by the receptive field of the feature extractor. The trade-off is that a small receptive field may not capture enough contextual information, while applying a large receptive field would make feature maps spatially coarse and make it hard to
Figure 2: Illustration of episodic training and (a) “learning by inpainting”. In each episode, a support set is constructed with normal samples. The features of the normal samples are plugged into the adaptive sparse coding layer as the dictionary. Synthesized abnormal samples are corrected by performing gradient corrections with the gradient obtained by the direction of energy minimization.

spot small anomaly regions. To solve this dilemma, instead of carefully tuning the receptive field of each layer of the feature extractor, we introduce a simple yet effective technique of applying the receptive field on the sparse coding layer. Specifically, as illustrated in Appendix Fig A, rather than performing sparse coding to each individual d-dim feature vectors, we apply it on $d \times l \times l$ volumes centered around each feature vector, where $l$ is the receptive field. This is equivalent to applying a $l \times l$, sliding window on spatial axes of the feature map and can be easily implemented by image to column (Im2Col) operation. Then we flatten the feature volumes into $dl^2$-dim vectors and adjust the shape of the dictionary accordingly. In this way, we are able to capture contextual information without needing to carefully tune the architecture of the feature extractor and we show in the later experiments that this technique improves the robustness of the network on different types of objects.

3.1.2 Shrinkage Function

The effectiveness of training the EBM for localizing anomaly regions heavily depends on the gradient propagation from later to earlier layers. It is shown in [15] that smooth activation functions like Swish [45] could be beneficial here. Notably, the gradients of the dictionary $D$ are determined by the sparse coding coefficients $\alpha$ as shown in Eqn. 6. However, the sparsity constraint of $\alpha$ would turn off the gradient computation of many elements in $D$ and this could be detrimental during the early stage of the training. To alleviate the sparse gradient issue, we replace the RELU-like shrinkage function in Eqn. 7 with its smoothed counterparts by introducing the Sigmoid based shrinkage functions (SigShrink). The SigShrink function is originally proposed for non-parametric signal estimation in [3], and can be defined as:

$$\sigma_\tau(z, \lambda) = \frac{z}{1 + \exp(-\tau(|z| - \lambda))},$$

where $\tau$ is the hyperparameter of smoothness. We present visualizations of the hard shrinkage function Eqn. 7 and SigShrink with different values of $\tau$ in Fig B. Comparing to the hard shrinkage function which truncates small values into zeros, the SigShrink with a large $\tau$ can sharply force small values to near-zeros. Therefore, the SigShrink will guarantee non-zero gradients everywhere.

3.2 Episodic Training

To train the proposed adaptive EBM, we perform episodic training that is widely adopted by meta-learning few-shot learners [18, 51]. Following the terminology of few-shot learning, in each training episode, the model is adapted and tested with a task sampled from the underlying task distribution. Specifically, the model is adapted to a support set of the given task, then a query set with ground truth labels is applied to evaluate the adaptation, which is used to update the model parameters. As shown in Fig 2, the support set of the $i$-th episode task contains a small number of $K$ normal samples $\{s^i_k\}_{k=1}^K$. The features $z^i_k = \Psi(x^i_k, \theta)$ of these normal samples are plugged into the dictionary $D^i \in \mathbb{R}^{d \times Kh'w'}$ corresponding to the $i$-th task to adapt the dictionary. After that, the adapted model
is measured by a query set consisting of $M$ normal samples $\{q^i_m\}_{m=1}^M$ and $M$ abnormal samples $\{\hat{q}^i_m\}_{m=1}^M$. Note that there is no actual abnormal samples given during training, instead, they are iteratively sampled from the EBM and the sampling will be discussed in details in Section 3.3. Recall that the training of EBM with contrastive divergence as in Eqn. 3 requires the estimation of energy scores of both positive samples from the true data distribution and negative samples from the modeled distribution. The positive energy can be estimated empirically with normal query set samples. The negative energy can be estimated by performing the MCMC-based sampling technique, typically Langevin Dynamics as described in Eqn. 4. Denoting the output of Langevin dynamics (sampled abnormal samples) initialized with $\hat{q}^i_m$ as $\text{LD}(\hat{q}^i_m)$, we have the empirical estimation of the contrastive divergence of the $i$-th episode as:

$$L_{cd} = \frac{1}{m} \sum_{m=1}^M \left[ E_{\theta}(q^i_m; D^i) - E_{\theta}(\text{LD}(\hat{q}^i_m); D^i) \right]. \quad (10)$$

With the energy score equivalent to the feature reconstruction error in Eqn. 8, minimizing $L_{cd}$ encourages normal features to be well-reconstructed by a sparse linear combination of dictionary atoms while the features from abnormal samples tend to produce relatively higher reconstruction errors so that they can be easily spotted.

### 3.3 Synthesizing Negative Samples

Typical EBM training with contrastive divergence conducts negative sampling from the modeled density using techniques such as Langevin Dynamics, which applies gradient descent to a noise initialization with small step size and large number of steps [17]. Such negative sampling steps can be costly and we argue that it is unnecessary in our case. Instead, we introduce a new strategy of “learning by inpainting”. Starting from a positive query sample $q^i_m$, we synthesize the corresponding negative sample $\hat{q}^i_m$ by randomly placing a small uniform noise patch on the image. The Langevin Dynamics procedure is then initialized with the resulting image instead of a noise image. As the Langevin Dynamics proceeds, synthesized abnormal samples $\text{LD}(\hat{q}^i_m)$ are inpainted along the direction of “normal”, $q^i_m$, and we introduce the following reconstruction loss:

$$L_{rec} = \frac{1}{m} \sum_{m=1}^M \text{MSE}(\text{LD}(\hat{q}^i_m), q^i_m). \quad (11)$$

We show in Fig 2(a) that, starting from a synthesized abnormal sample, only 5 steps of Langevin dynamic would be sufficient to make it visually close to the corresponding normal sample during training, serving as “hard negatives” that further facilitates the learning. The final loss of the episodic training is simply:

$$L = \eta_0 L_{rec} + \eta_1 L_{cd}, \quad (12)$$

where $\eta_0$ and $\eta_1$ are balance two loss terms. We summarize the overall training and inference procedure in Alg. 1 and Alg. 2 respectively.

### 4 Experiments

In this section, we conduct evaluation on the industrial inspection task with the MVTec-AD dataset [5, 6] (Section 4.1). Even though our proposed framework is image-based, we further demonstrate its efficacy on the video anomaly detection task in Section 4.2. In Section 4.3, we show ablations and insights relating to the adaptive sparse coding components. We show additional ablations including the superiority of using gradient of EBMs over pixel-wise reconstruction to localize anomalies in App. B and we provide implementation details in App. A.

#### 4.1 Industrial Inspection

The goal of this anomaly detection task is to predict whether a manufactured component contains any defects. The MVTec-AD dataset includes 15 categories of object. To demonstrate the fast adaptation capability of the proposed method, we adopt a leave-one-out training strategy. Specifically, samples of each target category are reserved for testing only, and the episodic training is performed on the
Algorithm 1 Training procedure.
1: Given: A feature extractor $\Psi$ with parameter $\theta$; a training dataset of multiple tasks with positive (normal) samples only.
2: Given: Number of shots $K$; number of query samples $Q$; step size $\beta$ of Langevin dynamics; total training episodes $I$; and learning rate $\epsilon$.
3: Initialize the feature extractor $\Psi$.
4: for Episode $i = 1 : I$ do
5: Sample the $i$-th task from the dataset, and randomly pick $K + M$ samples to form the support set $\{s^i_k\}_{k=1}^K$ and the query set $\{q^i_m\}_{m=1}^M$.
6: Generate corrupted query samples $\{q^i_m\}_{m=1}^M$ by placing random patches to $\{q^i_m\}_{m=1}^M$.
7: Extract the support and query sample features with $\Psi$ and update the adaptive sparse coding layer with $i$-th task dictionary $D^i$, which is constructed by support sample features. The energy function of the $i$-th task is now parametrized by $E_\theta(\cdot; D^i)$.
8: Obtain synthesized negative samples $\{L D(q^i_m)\}_{m=1}^M$ with the updated energy function using Langevin dynamic in (9).
9: Obtain the final loss $\mathcal{L}$ with $\mathcal{L}_{cd}$ (10) and $\mathcal{L}_{rec}$ (11).
10: Update parameters $\theta \leftarrow \theta - \epsilon \nabla_{\theta} \mathcal{L}$.
11: end for
12: Return $\Psi$ with parameter $\theta$.

Algorithm 2 Inference procedure on a task indexed by $i$.
1: Given: Feature extractor $\Psi$ with trained parameter $\theta$.
2: Given: The support set $\{s^i_k\}_{k=1}^K$ and query samples $\{q^i_m\}_{m=1}^M$ to be tested.
3: Extract normal feature tensor $Z^i \in \mathbb{R}^{d \times K \times h' \times w'}$.
4: Reshape the normal feature tensor into a matrix, and use it as the task-specific dictionary $D^i \in \mathbb{R}^{d \times K \times h' \times w'}$.
5: Estimate the anomaly score of a test sample $q^i_m$ by its energy score using Eqn (8), with $E_\theta(q^i_m; D^i) = \| \Psi(q^i_m; \theta) - D^i \cdot S(\Psi(q^i_m; \theta); D^i) \|^2$, where a higher energy score indicates that $q^i_m$ is more likely to be an abnormal sample.
6: The pixel-wise anomaly map of a test sample $q^i_m$ can be obtain by visualizing $\nabla q^i_m - E_\theta(q^i_m; D^i)$, where abnormal regions show higher magnitude.

remaining categories. During the training stage, the model will not see any samples from the target category. During testing, we first adapt the model to the target category with 10 randomly selected normal samples, then measure the performance with the entire testing set. We run the test 5 times, each time the model is adapted to random sets of 10 normal samples from the target category.

In Table 1 we first show performance of “upper-bound” methods, which train each category from scratch with massive normal samples. Specifically, [9] [10] train auto-encoders (AE) with normal samples and measure the reconstruction errors during the inference; AnoGAN [11] adopts a generative adversarial network (GAN) to learn the manifold of normal; VE-VAE [29] presents a visually explainable variational auto-encoder through gradient-based attention. For apple-to-apple comparison, we create a strong baseline by applying model-agnostic meta-learning (MAML) on an AE (denoted as MAML-AE, more details in App. Sec. A.3). To the best of our knowledge, the proposed method is the first that is capable of producing strong performance on new tasks, using just a single forward pass and no further training. This strongly suggests that the learned parameters are effectively shared across tasks, greatly helping to accelerate the model deployment process that is typically cumbersome otherwise. All results from our methods are obtained with flipping and 90° rotation as augmentations. The proposed method outperforms MAML-AE by a large margin. Our results are even competitive with the “upper-bounds” in some categories. We show the localized anomaly regions from our method in Fig 3. Additional visualizations are in the App. Fig D.

4.2 Video Surveillance

In video anomaly detection, a common goal is to detect abnormal events captured by surveillance cameras (e.g., a motorcycle on the sidewalk). A model trained on videos from one camera might not generalize well on other cameras due to different locations / mounting heights / lightning conditions, and it is not feasible to train one model for every new camera in practice. The ability to quickly adapt to new scenes is a significant contribution to the task of video surveillance. We are only aware of the
Figure 3: Visualizations of localized anomaly by our method.

Table 1: Numerical evaluation of anomaly localization on MVTec-AD. We report both mIoU (top rows) and AUC-ROC (bottom rows) values. Col 2-5 are upper-bound methods trained with massive normal samples.

work in \[32\] (r-GAN) that has such adaptation capability. Specifically, the model adapts to a new scene using gradient descent with several beginning frames of a query video, after which a GAN is applied to generate future frames. Anomaly is then detected via the discrepancy between predicted future frames and the original frames. Note that the MAML-AE baseline we conducted in Section 4.1 can be seen as an ablation of r-GAN on the single-frame without temporal information.

We follow the same evaluation regime as r-GAN by training with normal samples in all 13 scenes from SH-Tech \[28\] and testing on UCSD Pedestrian 1, UCSD Pedestrian 2 \[34\], and CUHK Avenue \[30\]. Note that since our method is image-based, it predicts the video frames independently without leveraging any temporal information as in r-GAN. In each episode, we adapt our model with a support set containing a few normal frames randomly sampled from the target scenes. In Table 2, we compare our method against r-GAN pre-trained on SH-Tech only (r-GAN Pre-train), fine-tuned on target datasets (r-GAN Fine-tune), and with one step gradient descent with meta-learning (r-GAN MAML). We also show the performance of MAML-AE as a baseline for image-based meta-learning method. In the last section of Table 2, we present intra-dataset results as well by training with 6 scenes of SH-Tech and testing on remaining 7. We follow common evaluation protocol and measure the frame-level AUC-ROC. Without leveraging temporal information and re-training (gradient descent), our method achieves comparable results to r-GAN MAML and outperforms image-based meta-learning method by a large margin. In App B.4, we show that incorporating simple temporal information can further improve the performance.

4.3 Ablation Studies

Sparse coding receptive fields. To evaluate the effectiveness of using large receptive fields in the sparse coding layer, we conduct additional experiments on the MVTec-AD dataset, and select 5 representative categories with different levels of difficulties to present the comparisons with \(l = 1\) and \(l = 3\) (Sec. 3.1.1) in Table 3. Sparse coding with large receptive field clearly benefits more complex
Table 2: Frame-level AUC-ROC for the video anomaly detection tasks.

<table>
<thead>
<tr>
<th>Target datasets</th>
<th>Methods</th>
<th>1-shot</th>
<th>5-shot</th>
<th>10-shot</th>
</tr>
</thead>
<tbody>
<tr>
<td>UCSD Ped 1</td>
<td>r-GAN Pre-train</td>
<td>73.10</td>
<td>73.10</td>
<td>73.10</td>
</tr>
<tr>
<td></td>
<td>r-GAN Fine-tune</td>
<td>76.99</td>
<td>77.85</td>
<td>78.23</td>
</tr>
<tr>
<td></td>
<td>r-GAN MAML</td>
<td>80.60</td>
<td>81.42</td>
<td>82.38</td>
</tr>
<tr>
<td></td>
<td>MAML-AE</td>
<td>64.12</td>
<td>66.88</td>
<td>67.34</td>
</tr>
<tr>
<td></td>
<td>Ours</td>
<td>77.42</td>
<td>78.12</td>
<td>78.65</td>
</tr>
<tr>
<td>UCSD Ped 2</td>
<td>r-GAN Pre-train</td>
<td>81.95</td>
<td>81.95</td>
<td>81.95</td>
</tr>
<tr>
<td></td>
<td>r-GAN Fine-tune</td>
<td>85.64</td>
<td>89.66</td>
<td>91.11</td>
</tr>
<tr>
<td></td>
<td>r-GAN MAML</td>
<td>91.19</td>
<td>91.80</td>
<td>92.80</td>
</tr>
<tr>
<td></td>
<td>MAML-AE</td>
<td>78.24</td>
<td>82.04</td>
<td>83.30</td>
</tr>
<tr>
<td></td>
<td>Ours</td>
<td>91.22</td>
<td>92.00</td>
<td>92.45</td>
</tr>
<tr>
<td>CUHK Avenue</td>
<td>r-GAN Pre-train</td>
<td>71.43</td>
<td>71.43</td>
<td>71.43</td>
</tr>
<tr>
<td></td>
<td>r-GAN Fine-tune</td>
<td>75.43</td>
<td>76.52</td>
<td>77.77</td>
</tr>
<tr>
<td></td>
<td>r-GAN MAML</td>
<td>76.58</td>
<td>77.10</td>
<td>78.79</td>
</tr>
<tr>
<td></td>
<td>MAML-AE</td>
<td>68.72</td>
<td>69.67</td>
<td>70.01</td>
</tr>
<tr>
<td></td>
<td>Ours</td>
<td>80.68</td>
<td>83.41</td>
<td>84.46</td>
</tr>
<tr>
<td>Sh-Tech</td>
<td>r-GAN Pre-train</td>
<td>70.11</td>
<td>70.11</td>
<td>70.11</td>
</tr>
<tr>
<td></td>
<td>r-GAN Fine-tune</td>
<td>71.61</td>
<td>70.47</td>
<td>71.59</td>
</tr>
<tr>
<td></td>
<td>r-GAN MAML</td>
<td>74.51</td>
<td>75.28</td>
<td>77.36</td>
</tr>
<tr>
<td></td>
<td>MAML-AE</td>
<td>66.62</td>
<td>67.12</td>
<td>68.04</td>
</tr>
<tr>
<td></td>
<td>Ours</td>
<td>75.32</td>
<td>79.64</td>
<td>81.28</td>
</tr>
</tbody>
</table>

Table 3: Comparison of different sparse coding receptive fields. We report both mIoU (left) and AUC-ROC (right) values.

<table>
<thead>
<tr>
<th>Category</th>
<th>Leather</th>
<th>Grid</th>
<th>Hazelnut</th>
<th>Cable</th>
<th>Zipper</th>
</tr>
</thead>
<tbody>
<tr>
<td>l = 1</td>
<td>0.40</td>
<td>0.95</td>
<td>0.11</td>
<td>0.80</td>
<td>0.36</td>
</tr>
<tr>
<td>l = 3</td>
<td>0.40</td>
<td>0.95</td>
<td>0.12</td>
<td>0.81</td>
<td>0.40</td>
</tr>
</tbody>
</table>

Table 4: Performance w/ and w/o sparsity constraint. From left to right: mIoU; AUC-ROC; the difference of averaged reconstruction errors between abnormal/normal samples.

<table>
<thead>
<tr>
<th>Category</th>
<th>Leather</th>
<th>Hazelnut</th>
<th>Cable</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ours</td>
<td>0.40</td>
<td>0.95</td>
<td>1.6e-4</td>
</tr>
<tr>
<td>No sparsity</td>
<td>0.32</td>
<td>0.90</td>
<td>0.9e-4</td>
</tr>
</tbody>
</table>

Figure 4: Visualizations of anomaly localization with video anomaly detection.

Figure 5: Loss curves with smooth (SigShrink) and non-smooth (hard-shrink RELU-like) shrinkage functions.

structural objects (hazelnut, cable, and capsule), while the improvements are limited for the texture objects (leather and grid), where contextual regularization is intuitively less important.

Shrinkage functions. To show the benefits of smooth shrinkage function, we plot the loss curves of models trained with smooth SigShrink (Eqn. 9) and non-smooth RELU-like shrinkage (Eqn. 7) functions in Fig 5. The model with smooth shrinkage function converges notably faster in the early training stage and achieves lower loss.

Sparsity constraint. As discussed in Section 3.1, we impose sparsity constraint to the feature decomposition in the adaptive sparse coding layer, in order to prevent abnormal features from being well-approximated by the linear combinations of normal features, so that the reconstruction errors are effective for detecting anomaly. To validate this, we conduct experiments by removing the shrinkage function $\sigma$ in the sparse coding stage (Eqn. 6). We show comparison in Table 4 with mIoU, AUC-ROC, and the difference of averaged reconstruction errors between abnormal and normal samples. Without sparsity, the performance drops dramatically, and reconstruction errors of normal and abnormal samples become closer.
5 Related Work

Anomaly detection with sparse coding. Early efforts on adopting sparse coding in anomaly detection are based on optimization (with L1 penalty) \[30, 61\]. Recent advances on iterative sparse thresholding algorithms \[11, 25\] allow seamless integration of online sparse coding with deep neural networks, and \[23\] formulates the sparse coding as stack RNNs for video anomaly detection.

Anomaly detection with generative models. One of the core challenges in anomaly detection is that abnormal samples are usually unavailable in the training stage. Generative models are widely utilized in anomaly detection due to the capability in modeling the density of desired data distribution. Early efforts on variational autoencoders (VAE) based methods \[11, 58\] are arguably having hard time calibrating uncertainties in novel samples \[36\], accurately localizing abnormal regions through reconstruction errors \[12\]. Recent efforts have explored variant generative architectures like energy-based models (EBM) \[20\], GANs \[50\], and combining VAE with EBM \[12\]. Various methods also exploit intra-image structures \[10, 7\], cross-frame consistency \[31\], and motion-appearance consistency in videos \[39\] while detecting anomaly.

Few-shot learning. Few-shot learning is extensively explored in classification tasks. It leverages common knowledge extracted from a distribution of tasks, and induces an adaptive model that fits to a new classification task with as few as one sample per class. Proposed methods are based on optimization \[18, 42, 12, 59, 46\], learning metric \[51, 53\] and parameter prediction \[22, 43, 21\]. These technologies are further applied in other tasks like image generation \[9, 27\] and out-of-distribution detection \[49\].

Energy-based models. As a family of generative models, studies on EBMs \[26\] are mainly focused on probabilistic modeling and sampling over data, either unconditionally \[38, 44, 40, 17, 13, 63, 57, 56, 2\], or conditionally \[13, 14\]. This has been recently followed by extensions to other applications that include reasoning \[16\], latent space modeling of generative models \[41\], and anomaly detection \[12\].

6 Conclusion

In this paper, we introduced few-shot fast-adaptive anomaly detection. We formulated our model as an energy-based model with an adaptive sparse coding layer, of which the dictionary is directly formed by normal features of a target task. We adopted episodic meta-learning to learn common knowledge across tasks, which enables few shots adaptation. We further introduced smooth shrinkage functions, sparse coding with large receptive fields, and learning by inpainting to improve and accelerate the training. Notably, when evaluating our method’s performance on industrial inspection and video anomaly detection, our method is comparable and even boasts better performance than methods trained with a large amount of normal samples. Through this work, we hope to have made a significant contribution to the important problem of anomaly detection by shedding light on our findings that anomaly detection can indeed be generalized to new tasks with a few normal samples only.

Social Impact and Ethics. As a general framework for few-shot anomaly detection, the proposed method does not suffer from particular ethical concerns or negative social impacts. All datasets used are public, and we have blurred all human faces in the qualitative visualizations.
References


[38] Jiquan Ngiam, Zhenghao Chen, Pang W Koh, and Andrew Y Ng. Learning deep energy models. In ICML, 2011.


Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper’s contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [No]
   (c) Did you discuss any potential negative societal impacts of your work? [N/A]
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]

2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [N/A]
   (b) Did you include complete proofs of all theoretical results? [N/A]

3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [No]
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes]
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [No]
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes]

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes]
   (b) Did you mention the license of the assets? [N/A]
   (c) Did you include any new assets either in the supplemental material or as a URL? [No]
   (d) Did you discuss whether and how consent was obtained from people whose data you’re using/curating? [No]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [Yes]

5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]