Spatially Constrained Adversarial Attack Detection and Localization in the Representation Space of Optical Flow Networks

Hannah Kim[1,2*], **Celia Cintas**[2], **Girmaw Abebe Tadesse**[2†] and **Skyler Speakman**[2]
[1]Duke University
[2]IBM Research Africa
hannah@cs.duke.edu, {celia.cintas, girmaw.abebe.tadesse}@ibm.com, skyler@ke.ibm.com

Abstract

Optical flow estimation have shown significant improvements with advances in deep neural networks. However, these flow networks have recently been shown to be vulnerable to patch-based adversarial attacks, which poses security risks in real-world applications, such as self-driving cars and robotics. We propose SADL, a Spatially constrained adversarial Attack Detection and Localization framework, to detect and localize these patch-based attack without requiring a dedicated training. The detection of an attacked input sequence is performed via iterative optimization on the features from the inner layers of flow networks, without any prior knowledge of the attacks. The novel spatially constrained optimization ensures that the detected anomalous subset of features comes from a local region. To this end, SADL provides a subset of nodes within a spatial neighborhood that contribute more to the detection, which will be utilized to localize the attack in the input sequence. The proposed SADL is validated across multiple datasets and flow networks. With patch attacks 4.8% of the size of the input image resolution on RAFT, our method successfully detects and localizes them with an average precision of 0.946 and 0.951 for KITTI-2015 and MPI-Sintel datasets, respectively. The results show that SADL consistently achieves higher detection rates than existing methods and provides new localization capabilities.

1 Introduction

Motion estimation is a key problem in computer vision. Motion in video is often represented in the form of dense optical flow fields, which specify the motion of each pixel from one frame to the next. It has diverse application areas, such as tracking [Shi and Tomasi, 1994], action recognition [Diba *et al.*, 2017], 3D reconstruction [Agarwal *et al.*, 2009], video interpolation [Kim *et al.*, 2022b], and video compression [Le Gall, 1991].

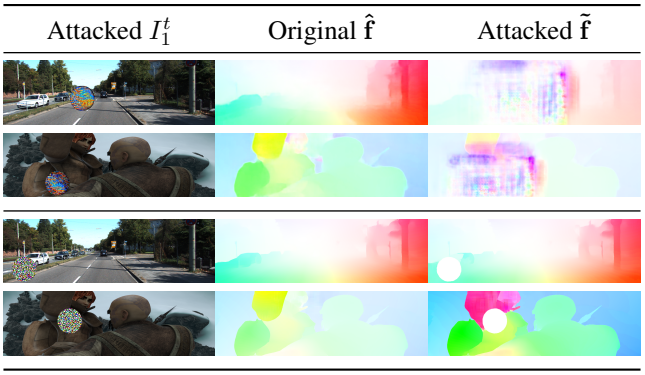

Figure 1: Effect of 153×153 patch attacks on flow estimations from FlowNetC (top) and RAFT (bottom) using KITTI 2015 and MPI-Sintel datasets.

With the emergence of deep learning techniques and availability of large scale datasets, optical flow estimation performance has been significantly improved [Dosovitskiy *et al.*, 2015; Ilg *et al.*, 2017; Sun *et al.*, 2018; Teed and Deng, 2020] across existing benchmarks [Butler *et al.*, 2012; Menze and Geiger, 2015]. However, recent work [Ranjan *et al.*, 2019; Schrodi *et al.*, 2021; Schrodi *et al.*, 2022] demonstrated that these networks are also vulnerable to adversarial attacks just as any other deep learning approaches [Papernot *et al.*, 2017; Akhtar and Mian, 2018; Akinwande *et al.*, 2020; Kim *et al.*, 2022a], thereby raising security issues, e.g., in self-driving cars. These attacks on flow estimators come in the form of patches that are pasted onto the input image frames at the same pixel coordinates. This is motivated by cases where something gets stuck on the video camera (see Figure 1). These patch-based attacks cause occlusions on areas where the attack is placed, and motion boundaries along the boundaries of the attack, which is where current flow estimators suffer [Kim *et al.*, 2021; Yu *et al.*, 2022].

Schrodi *et al.* [2019] is the first work to tackle optical flow estimators with adversarial patches that are optimized so that the direction of the original flow estimations are shifted towards the opposite direction. They continue this work to show that the location these adversarial patches are applied on

*This work was done when Hannah Kim was an intern at IBM Research. She now works at Amazon.

†This work was done when Girmaw was at IBM Research. He now works at Microsoft. Email: gtadesse@microsoft.com

the input images is important [Schrodi *et al.*, 2021; Schrodi *et al.*, 2022]. Figure 1 shows the effect of these patch attacks on two flow estimators, FlowNetC [Dosovitskiy *et al.*, 2015] (top) and RAFT [Teed and Deng, 2020] (bottom), the current state of the art on flow estimation. While RAFT is more robust to these attacks than FlowNetC, these attacks still disrupt the flow estimations nearby these attacks, e.g. estimations on moving white car and on arm and face. Schrodi *et al.* [2021; 2022] further explain that the vulnerability of flow estimators comes from the attacks' self-similar patterns. However, there is no existing work, to the best of the authors' knowledge, that was proposed to detect, localize, and characterize these potential attacks.

We present a Spatially constrained adversarial Attack Detection and Localization (SADL) framework that aims to detect and localize patch-based attacks given any pre-trained flow estimators in an unsupervised fashion without prior knowledge about these attacks. Specifically, we propose analyzing the inner layers' activations with a local-neighborhood constraint across the spatial dimension.

The main contributions of this paper are:

- First to detect and localize patch-based adversarial attacks for optical flow networks based on spatially constrained subset scanning on the inner layer activations, without any training of a dedicated model nor prior knowledge of the patch-based attacks.
- Analysis of the effect of the attacks across various inner layers of the flow networks.
- Validate our method across several state-of-the-art optical flow networks and datasets.

2 Related Work

In this Section, we reviewed the related work on flow networks, adversarial attacks on flow networks, and subset scanning techniques from anomalous pattern detection literature.

Flow networks Recent methods [Sun *et al.*, 2018; Xu *et al.*, 2017; Dosovitskiy *et al.*, 2015; Teed and Deng, 2020] on optical flow estimation employed deep learning to achieve promising results with a faster inference speed. Starting with Chen and Koltun [2016], these deep learning approaches used cost volumes [Rhemann *et al.*, 2013], which comprised of feature correspondence scores between frames to compute optical flow. PWCNet [Sun *et al.*, 2018] built the cost volumes at multiple scales, and FlowNet [Dosovitskiy *et al.*, 2015] included the full cost volume at a single scale. Ilg *et al.* [2017] proposed FlowNet2 by stacking four FlowNet-based models. Teed and Deng [2020] achieved state-of-the-art performance with RAFT using a four-dimensional cost volumes for all pairs of pixels on the input images.

Adversarial attacks in optical flow neural networks Ranjan *et al.* [2019] is the first work to apply adversarial attacks on optical flow models, by pasting an $p \times p$ adversarial patch onto the input image frames I_1, I_2 at the same location. This adversarial patch can be obtained using any pre-trained flow network by minimizing the cosine similarity between the estimated flow maps before $\hat{\mathbf{f}} = (u, v)$ and after the attack $\tilde{\mathbf{f}} = (\tilde{u}, \tilde{v})$, i.e., $\hat{\mathbf{f}} \cdot \tilde{\mathbf{f}} / ||\hat{\mathbf{f}}|| ||\tilde{\mathbf{f}}||$. This work was later extended by Schrodi *et al.* [2021] to show that patch-based attacks trigger matching ambiguities in the correlation output, which are successively spread in the decoder into a wider neighborhood. In their subsequent work, Schrodi *et al.* [2022] demonstrated the importance of the spatial location that these adversarial patches applied on the input image frames. Large flow regions, e.g., fast-moving objects, are susceptible to a severe deterioration of flow estimations. Yamanaka *et al.* [2021] used patch-based adversarial attacks to simultaneously attack two CNNs developed for optical flow and monocular depth estimation. Wortman [2021] showed that these patch attacks for optical flow estimation can be hidden by adjusting their alpha values while still affecting the flow estimations. Global noise-based attacks were recently reported [Schrodi *et al.*, 2022; Schmalfuss *et al.*, 2022]. Schrodi *et al.* showed that optical flow networks are robust to white-box global noise-based attacks, and Schmalfuss *et al.* [2022] proposed perturbation constrained flow attack. Unlike the existing works, we focus on detecting the patch-based attacks in an unsupervised fashion without requiring any prior knowledge of the attacks. Our work further localizes these patch attacks in the input images, finding the origin of the attack, which can be useful in future mitigation efforts. To the best of the author's knowledge, we are the first work to detect or localize adversarial attacks on optical flow estimators.

Subset Scanning for Anomalous Pattern Detection in Neural Networks Subset scanning has been used to detect anomalous samples in various computer vision and audio tasks, including creativity sample characterization [Cintas *et al.*, 2022], audio adversarial attacks [Akinwande *et al.*, 2020] and skin condition classification [Kim *et al.*, 2022a]. To detect anomalous samples, subset scanning searches for the "most anomalous" subset of node activations from the inner layers of the given test sample on any pre-trained networks, and uses these subset of nodes to obtain the anomalousness score of the test sample.

While existing works employed subset scanning to detect adversarial attacks, we are the first to use it to detect patch-based attacks as opposed to global noise-based attacks, on sequential images across temporal dimension, and on networks for regression instead of classification task. Since patch attacks in flow estimation occur in a local region, we propose to apply spatial constraints to the scanning step to improve the detection and localization of the attacks.

3 Proposed Approach: Spatially Constrained Subset Scanning

Given node activations of an inner-layer, $\mathcal{L}$, of a pre-trained optical flow network Θ, we aim to detect a subset of test samples with adversarial attacks and localize these attacks. To this end, we propose SADL: Spatially constrained Adversarial attack Detection and Localization framework (see Fig. 2) to be applied on any pre-trained flow estimators, in an unsupervised fashion without any prior knowledge of the attacks.

3.1 Overview

Figure 2 illustrates an overview of our proposed method. We extract the node-level activations across a given inner-layer

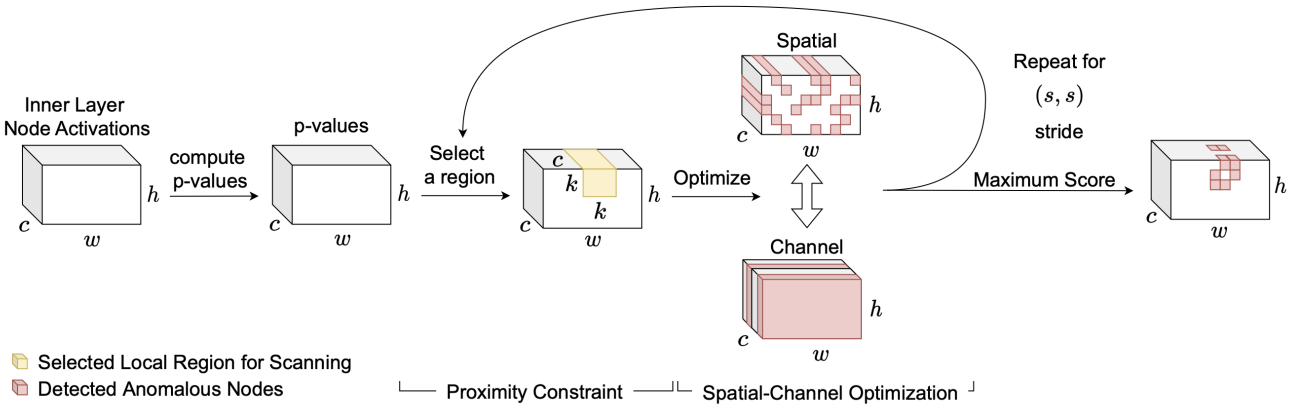

Figure 2: Overview of our spatially constrained subset scanning method for adversarial patch-attack detection and localization.

of the flow network for a pair of input images. We compare this *observed* activation against an empirical distribution of *expected* activations extracted from clean (un-attacked) images at that same node. This node-level comparison provides a p-value for each node's *observed* activation. If the test image pair has the same representation as un-attacked images in this flow network layer, then these observed p-values across nodes should appear *uniformly* distributed. However, if an adversarial attack systematically increases the activations, then the observed p-values would appear non-uniform.

Our scoring function, to quantify the deviation of the observation from the expectation, is a non-parametric goodness-of-fit statistic which measures the Kullback–Leibler divergence between the observed distribution of p-values and the expected, uniform distribution [Berk and Jones, 1979]. A high-scoring image pair suggests it has a different inner-layer representation than the clean, un-attacked images. Critically, this scoring function satisfies the Linear-time subset scanning (LTSS) property [Neill, 2012; McFowland *et al.*, 2013] which allows for exact, efficient maximization over *subsets* of nodes. However, *un*-constrained maximization across all subsets of nodes may lead to high scores even when a test pair of images is not attacked (false positive due to matching to noise). Therefore, SADL implements *channel constraints* and *proximity constraints* which result in higher detection power and accuracy for localized patch attacks. The end result of SADL is a high-scoring subset of node activations in a local, spatial neighborhood and subset of channels. The score is used for detection power calculations, while the subset of spatial nodes is used for detection accuracy. Details on the constrained optimization process are provided below.

3.2 p-value Computation

Let D_{H_0} be a dataset of M background images that have not been attacked. Feeding the background image pairs into Θ, we can extract M background activations $A_j^{H_0}$ from each node $O_j \in \mathcal{L}$. Given a test sample $I \notin D_{H_0}$, we extract the activation a_j at node O_j in $\mathcal{L}$. We can compute the p-value of the activation a_j of this test sample by computing the proportion of activations in $A_j^{H_0}$ that are greater than a_j:

$$p_j = \frac{\sum_M [A_j^{H_0} \geq a_j] + 1}{M + 1} , \tag{1}$$

where $[\cdot]$ is the indicator function.

3.3 Spatial Alignment

The inner layer under consideration, $\mathcal{L}$, has spatial dimensions w (width), h (height) and channels c. In order to properly incorporate spatial information across channels, we make the alignment assumption that if spatial location $w' \times h'$ is to be evaluated, then activations (p-values) from all channels under consideration also located at w' and h' must be scored as well. This intuitive assumption is visualized at the top of Figure 2. Although spatial alignment is always enforced, SADL does not require *all* channels to be included in the returned subset. This extension is covered in the Spatial-Channel Optimization section below.

3.4 Non-parametric Scan Statistics

Given a tensor of p-values, P, computed with Eq. 1 and with the same $w \times h \times c$ dimensions as the inner layer $\mathcal{L}$, our goal is to find the subset of nodes, aligned at spatial locations, such that the corresponding p-values of these node activations is the *most non-uniform*. We quantify divergence away from the uniform distribution using a non-parametric scan statistic (NPSS), such as Berk-Jones [Berk and Jones, 1979], with a general form:

$$F(S) = \max_{\alpha} \phi(\alpha, N_\alpha(S), N(S)) , \tag{2}$$

where S is the subset p-values (nodes) under consideration, $N(S)$ is the number of p-values in S, $N_\alpha(S)$ is the number of significant p-values less than a threshold α (*OptimizeNPSS*). NPSS is preferred as we make minimal assumptions on how the activations at a given node are distributed.

Without spatial alignment enforced, NPSS can be trivially maximized by creating a subset consisting of all-and-only p-values that are less than α. However, this naive maximization would likely violate the spatial alignment requirement by not necessarily including all p-values from the same spatial location. Enforcing spatial alignment makes this maximization

Algorithm 1: Scanning over p-values P to find the highest-scoring subset of features within a $k \times k$ spatial neighborhood and for some subset of channels.

input : $P \in [0,1]^{w \times h \times c}$
output: $F_{\max}$, $\mathbf{r}_{\max}$, $\mathbf{c}_{\max}$

$F_{\max} \leftarrow -1$;
for h_k *in* H = `LinearSpace(0, h, k)` **do**
  for w_k *in* W = `LinearSpace(0, w, k)` **do**
    $P_k \in [0,1]^{kk \times c} \leftarrow$ `FlattenSpatial(`$P[h_k : h_k + k, w_k : w_k + k, :]$`)` ;
    $\mathbf{r}_k \leftarrow$ `RandomlySelectRows(`P_k`)` ;
    $\mathbf{c}_k \leftarrow$ `RandomlySelectRows(`P_k^T`)` ;
    $F \leftarrow -1$;
    while F *is increasing* **do**
      $F, \mathbf{r}_k \leftarrow$ `OptimizeNPSS(`$P_k[:, \boldsymbol{c}_k]$`)` ;
      $F, \mathbf{c}_k \leftarrow$ `OptimizeNPSS(`$P_k[\boldsymbol{r}_k, :]^T$`)` ;
    if $F_{\max} < F$ **then**
      $F_{\max}, \mathbf{r}_{\max}, \mathbf{c}_{\max} \leftarrow F, \mathbf{r}_k, \mathbf{c}_k$;
return $F_{\max}, \boldsymbol{r}_{\max}, \boldsymbol{c}_{\max}$

more difficult because some not-significant p-values may be included in the highest-scoring spatially aligned subset, and some significant p-values may be excluded.

To this end, we employ the Linear-time Subset Scanning (LTSS) property of the Berk-Jones scan statistic [Neill, 2012; McFowland *et al.*, 2013]. Returning to the dimensions of the tensor, P, each of the wh spatial locations contains up to c p-values. There are $2^{wh} - 1$ possible subsets of spatial locations to consider with each location contributing c p-values to the scoring function. The LTSS property reduces this search space from exponential, $O(2^{wh})$ to linear $O(wh)$ while still guaranteeing exactness [Neill, 2012]. To do this, each spatial location is sorted by a priority function, $\frac{N_\alpha}{c}$. A spatial location with a priority of 1 means every one of the p-values at that location across the c channels is significant at level α. A priority value of 0.5 means only half of the p-values at that spatial location are less than α. Once the wh locations are sorted by this priority, the LTSS property states that the highest-scoring subset is guaranteed to be one of the linearly-many subsets created by incrementally including the c p-values of the next-highest priority location. Critically, the LTSS property turns the Berk-Jones from a metric that measures divergence into an objective scoring function that can be efficiently maximized over exponentially-many subsets. SADL, in its base form, searches over the exponentially many subsets of spatial locations, Ψ, to find spatially-aligned subset of p-values, S^*, with the highest score, F^*:

$$F^* = F(S^*) = \max_{S \in \Psi} F(S) \tag{3}$$

where $F(S)$ is a NPSS that satisfies the LTSS property such as the Berk-Jones scan statistic.

3.5 Spatial-Channel Optimization (SCO)

The base version of SADL optimizes a scoring function over the $2^{wh} - 1$ possible subsets of spatial locations in a tensor with $w \times h \times c$ dimensions and is forced to include *all* c channels in the returned subset. See the top part of Figure 2 for a visual example. This forced inclusion of all channels naively implies that all channels in the inner layer being scanned are impacted by a patch attack on an optical flow neural network. The more complicated reality is that only some *subset* of the c channels may be affected by the attack. By introducing Spatial-Channel Optimization features to SADL we are returning a subset of spatial locations $\mathbf{r}_{\max}$ *crossed with* a subset of channels $\mathbf{c}_{\max}$. Without SCO in place, SADL is forced to return a subset of spatial locations spanning *all* channels.

Including SCO in the search process can be done through two additional insights. The first is recognizing that the same process that maximizes a scoring function over subsets of spatial locations can also be used to maximize over a subset of channels. In other words, the LTSS property can be applied across a different mode of the tensor, P. The second is using an iterative ascent procedure that optimizes over spatial locations first, then over channels, and then back to spatial locations. Each step of the ascent is conditioned on the highest-scoring subset found so far, and this ascent is guaranteed to converge to a local maxima such that any change to either the subset of spatial locations or a subset of channels would *decrease* the score [McFowland *et al.*, 2013]. Two random restarts are used to approach the global maximum.

3.6 Proximity Constraint (PC)

While SCO allows us to find an anomalous subset of node activations across spatial locations, these detected anomalous locations may be far apart from each other, spanning the entire frame of the given test image pairs. We want to further constrain the subset scanning so that the detected anomalous locations strictly come from a local spatial neighborhood. Thus, we enforce a proximity constraint where our optimization is only applied to the $k \times k$ spatial region (yellow region in Figure 2) within the given $h \times w \times c$ p-values where $k << h$ and $k << w$ (see Algorithm 1). This will ensure that the detected subset of anomalous locations only comes from this $k \times k$ spatial neighborhood. We do this for all $k \times k$ locations in the given $h \times w$ locations with a stride of $s = k/2$, and return the subset of p-values from a $k \times k$ neighborhood that yields the highest NPSS score $F_{\max}$.

3.7 Patch Attack Localization

Given the subset of detected p-values from a $k \times k$ neighborhood with F_{max}, we can easily localize the attack by finding which locations these detected p-values occur in the inner layer feature map. These detected locations in the feature maps can be up-sampled, if needed, to match the resolution of the input images, or vice versa. For instance, if a pixel at location $(10, 10)$ is detected as anomalous in a feature map of size $20 \times 20 \times c$ under consideration, then SADL predicts 3×3 attack to be in the input images of size $60 \times 60 \times 3$ centered at location $(30, 30)$. The localized attacks can be utilized for attack mitigation, which we leave for future work.

4 Experimental Setup

Flow Networks We validate our approach using four state-of-the-art flow estimators, namely, FlowNetC [Dosovitskiy

Network	FlowNetC			FlowNet2			PWCNet			RAFT		
Attack Size p	0	153	51	0	153	51	0	153	51	0	153	51
KITTI 2015	11.50	38.85	31.60	10.07	12.24	12.09	12.55	17.01	16.27	5.86	8.85	7.37
MPI-Sintel	3.18	42.45	29.64	2.22	3.10	2.71	3.98	5.37	4.77	1.663	2.85	2.23

Table 1: Effect of adversarial patch attacks as measured by end-point error on four flow estimators for KITTI 2015 and MPI-Sintel. We use $p \times p$ adversarial patches ($p \in [153, 51]$) to attack our test images ($p = 0$ show original performances without any attacks).

et al., 2015], FlowNet2 [Ilg *et al.*, 2017], PWCNet [Sun *et al.*, 2018], and RAFT [Teed and Deng, 2020]. When selecting inner layers to apply our proposed method, we choose the first layer in each of the network's components. Specifically, we select the first layer of the encoder and decoder module, and their correlation layer. FlowNet2 contains four FlowNet-based models and a fusion module, and we apply our proposed subset scanning on the first layer in each of the components. For PWCNet, we select the first layer from the encoder, the cost volume layer and the optical flow estimator. Finally, for RAFT, we select the first layer in the feature encoder and context encoder, and correlation and flow layer in the iterative update block. For layers that process multiple features, (e.g. Siamese encoder of RAFT), we concatenate the features across the channel dimension and apply subset scanning on the concatenated features.

Datasets Following the literature [Schrodi *et al.*, 2022; Teed and Deng, 2020], we use KITTI 2015 [Menze and Geiger, 2015], raw KITTI [Geiger *et al.*, 2013], MPI-Sintel [Butler *et al.*, 2012], and raw Sintel [Liu *et al.*, 2019] datasets. **KITTI** consists of road scene images with sparse optical flow labels (2015) and without labels (raw). **MPI-Sintel** contains 23 sequences from computer-animation short "Sintel" with flow labels. Its raw frames without labels have also been used in previous un- or semi-supervised flow estimators [Liu *et al.*, 2019; Yuan *et al.*, 2022; Yuan *et al.*, 2023].

Generating patch-based attacks Following Ranjan *et al.* [2019] and Schrodi *et al.* [2022], we construct patch-based adversarial attacks on the four flow networks. Table 1 shows the change in end-point error with and without the adversarial patch attack. We use patches of four different sizes ($p \times p$) in respect to the input image resolution, 4.8% ($p = 153$), 2.1% ($p = 102$), 0.5% ($p = 51$), 0.1% ($p = 25$). These patches are trained using KITTI raw and Sintel raw datasets for each network, and evaluated on the labeled test set of KITTI 2015 and on MPI-Sintel, respectively. As expected, we see worse performances as we increase the patch attack sizes. In terms of flow networks, these patch attacks harm the performance of FlowNetC the most and RAFT the least (see Figure 1).

4.1 Performance Evaluation

We use the labeled data of Sintel and KITTI datasets to evaluate the attack detection and localization method. For KITTI 2015 dataset, we use the test split of its data as the training split has been used in training the existing flow networks. The test split of KITTI 2015 consists of 200 image pairs, and we use 100 of these image pairs as background image pairs ($I_1^{H_0}$ and $I_2^{H_0}$) to obtain our expected distribution of activation, the other 100 sample pairs as our test images (I_1^t and I_2^t), where 50 are non-attacked clean image pairs and the other 50 are anomalous image pairs with the patch attacks. Similarly, for Sintel dataset, we randomly select 200 image pairs for evaluation, 100 image pairs for background image pairs ($I_1^{H_0}$ and $I_2^{H_0}$), 50 sample pairs for anomalous test samples, and the last 50 sample pairs for clean test samples (I_1^t and I_2^t).

To evaluate the detection power of adversarial attacks, we follow the subset scanning in neural networks literature [Cintas *et al.*, 2020; Kim *et al.*, 2022a] and use the area under the receiver operating characteristic curve (AUC). For localization of the attacks, we report average precision (AP) and average recall (AR), where we compute precision and recall of the correctly detected patch attack locations for each sample and average over the entire test set. Here, the correctly detected locations occur when the detected anomalous locations on the given inner layer feature overlap with the true attacked location in the input space, down-sampled to match the resolution of the feature.

5 Results

In this section, we show our results on the detection and localization of adversarial patch attacks on optical flow estimation. As SADL is the first work to detect and localize these patch attacks, we use a naive scanning method without any spatial constraints as the baseline method.

Patch Attack Detection We show in Table 2 the performance of patch attack detection in terms of AUC comparing the proposed SADL (ours) with the baseline ([Cintas *et al.*, 2020]). Overall, we see higher detection performance with our proposed method (bolded). Figure 3 visualizes the distribution of F_{max} of the clean (blue) and attacked (orange) test set corresponding to the values in Table 2 for a patch attack of size $p = 153$ on KITTI 2015. The two distributions are more separable using SADL than the baseline. While our method shows clear improvements across multiple networks and datasets, both methods suffer on MPI-Sintel for FlowNetC and on KITTI 2015 for PWCNet. We show in Section 6 that the hyper-parameter k can be optimized to improve their performance.

Patch Attack Localization Table 3 lists the performance of patch attack localization in terms of average precision and average recall (see Section 4). As expected, we see higher localization performances for larger patches. While our method sometimes fails to localize patch attacks of size $p = 51$, these small patch attacks comprise 0.5% of the input images and do not have much effect on the flow estimation performance as shown in Table 1. Similar to attack detection re-

	Network	FlowNetC		FlowNet2		PWCNet		RAFT	
	p	153	51	153	51	153	51	153	51
KITTI 2015	[Cintas *et al.*, 2020]	0.50	0.51	0.55	**0.59**	0.53	**0.54**	0.52	0.58
	SADL (ours)	**0.98**	**0.78**	**0.72**	0.57	**0.58**	0.52	**1.00**	**0.61**
MPI-Sintel	[Cintas *et al.*, 2020]	0.50	0.54	0.64	**0.59**	0.66	**0.63**	0.54	0.53
	SADL (ours)	**0.58**	**0.77**	**1.00**	0.56	**0.88**	0.62	**1.00**	**0.61**

Table 2: Performance (AUC) of patch-based attack detection using subset scanning on FlowNetC, FlowNet2, PWCNet, and RAFT using a KITTI 2015 and a MPI-Sintel dataset compared to the baseline subset scanning method without spatial constraint. We list the performance of two sizes of patch attacks p, i.e., $p = 153$ and $p = 51$. Bold is the best in each column.

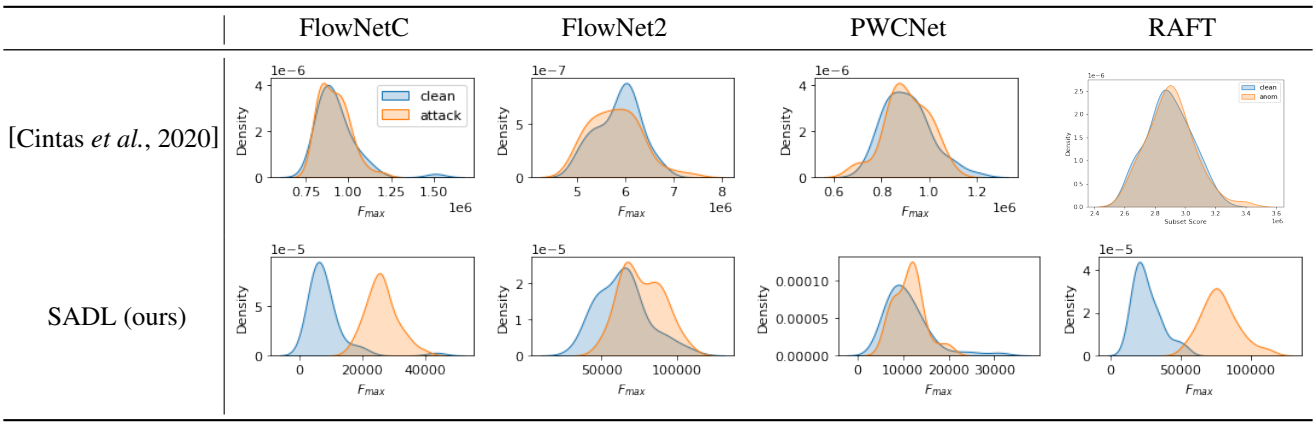

Figure 3: Distribution of anomalous scores obtained from the clean (blue) and the attacked (orange) test set with 153×153 patch attacks on FlowNetC, FlowNet2, PWCNet, and RAFT using the KITTI 2015 corresponding to the results in Table 2. The two distributions are more separable with our proposed method. See appendix for the distributions of other attack sizes and on MPI-Sintel.

sults above, SADL struggles to localize attacks on KITTI 2015 for PWCNet and on MPI-Sintel for FlowNetC, partly because our hyper-parameters are not optimized (see Section 6). Figure 5 shows an example visualization of a detected subset of anomalous locations in the feature space for the attacks of size $p = 153$ for KITTI 2015 (top) and MPI-Sintel (bottom). SADL successfully detects a subset of anomalous locations that align with the location of the patch attack.

6 Ablation Study

Performance across Layers Figure 4 shows the change in AUC for detecting 153×153 patch attacks as we go deeper in the four flow networks on KITTI 2015 dataset. For FlowNetC, we see better detection power as we go deeper into the network. This supports the claim made by Schrodi *et al.* [2022] that states that FlowNetC amplifies the effect of the patch attack going deeper into the decoder layers, making it easier to detect these anomalous behaviors in the deeper layers. The flow estimation by FlowNetC in Figure 1 also shows this pattern, where the effect of the patch attack is amplified to its neighborhood. For the other three networks, FlowNet2, PWCNet, and RAFT, we don't see this amplifying effect in the prediction, and they show better detection power in the earlier layers of the network. This observation aligns with Cintas *et al.* [2020], which shows that adversarial attacks can be better detected in the earlier layers of the network before

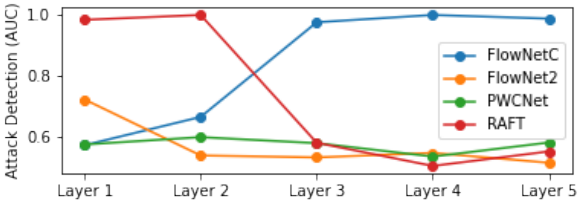

Figure 4: Detection performance (AUC) of 153×153 patch attacks across five inner layers as we go deeper into FlowNetC (blue), FlowNet2 (orange), PWCNet (green), and RAFT (red) on KITTI 2015. See appendix for the results of other patch sizes and dataset.

their effects get saturated in the deeper layers.

Performance without the Proposed Components We show the performance changes in detection (Table 4) and localization (Fig. 5), without our proposed components, i.e., proximity constraint (PC) and spatial-channel optimization (SCO). Without PC, the detected subset of anomalous locations could be spatially far apart, which would make it hard to localize the patch attacks in the inputs. Without SCO, the anomalous score F_{max} are computed using the activation across the entire channel in the detected locations, including the channels that are not anomalous. Overall, the use of the proposed components (SCO+PC) is shown to achieve the highest detection performance across different networks

	FlowNetC				FlowNet2				PWCNet				RAFT			
	$p = 153$		$p = 51$		$p = 153$		$p = 51$		$p = 153$		$p = 51$		$p = 153$		$p = 51$	
	AP	AR	AP	AR	AP	AR	AP	AR	AP	AR	AP	AR	AP	AR	AP	AR
KITTI 2015	0.95	0.35	0.63	0.73	0.38	0.27	0.01	0.05	0.33	0.19	0.02	0.07	0.95	0.66	0.01	0.04
MPI Sintel	0.02	0.00	0.50	0.73	0.92	0.66	0.04	0.09	0.66	0.29	0.19	0.38	0.95	0.70	0.18	0.34

Table 3: Performance of patch attack localization (AP/AR) of our proposed method across the four optical flow networks we consider.

and patch sizes. Figure 5 further demonstrates the capability of the proposed PC component to successfully find the most anomalous local region where the attack occurs.

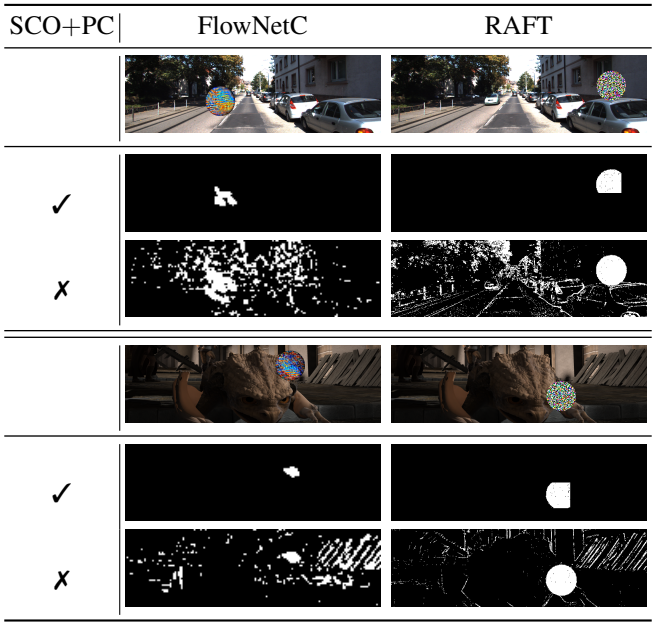

Figure 5: Example of the detected subset of anomalous locations (white) for the FlowNetC and RAFT on KITTI 2015 (top) and MPI-Sintel (bottom). In each panel, top row shows the attacked images and the bottom row shows the detected locations from SADL with and without the proposed components, proximity constraint (PC) and spatial-channel optimization (SCO).

Impact of Local Region Size for Proximity Constraint Figure 6 shows the effect of k used in our proximity constraint on the AUC of attack detection on FlowNetC. Specifically, the x-axis of the plots show the ratio of the height of the feature we are considering and k, i.e. h/k, and the y-axis shows AUC for attack detection for KITTI 2015 (left) and MPI-Sintel (right). Generally, we see increasing detection performance as we decrease k. We selected $k = h/3$ for results in Section 5 to be similar to the largest patch attack size ($p = 153$). However, we can achieve even higher attack detection performances with other k values, especially for MPI-Sintel with $p = 153$ patch attacks (cyan in the right plot). However, using smaller k will sacrifice recall performances in attack localization as the detected $k \times k$ region will be smaller than the patch attack.

Dataset	SCO+PC	FlowNetC		RAFT	
		$p = 153$	$p = 51$	$p = 153$	$p = 51$
KITTI 2015	✓	**0.98**	**0.78**	**1.00**	**0.61**
	✗	0.89	0.60	0.90	0.51
MPI-Sintel	✓	**0.58**	**0.77**	**1.00**	0.61
	✗	0.52	0.67	0.96	**0.65**

Table 4: Changes in AUC for patch-attack detection without the proposed components, i.e., proximity constraint (PC) and spatial-channel optimization (SCO), on KITTI 2015 and MPI-Sintel. Our method with all the components performs the best overall (bolded).

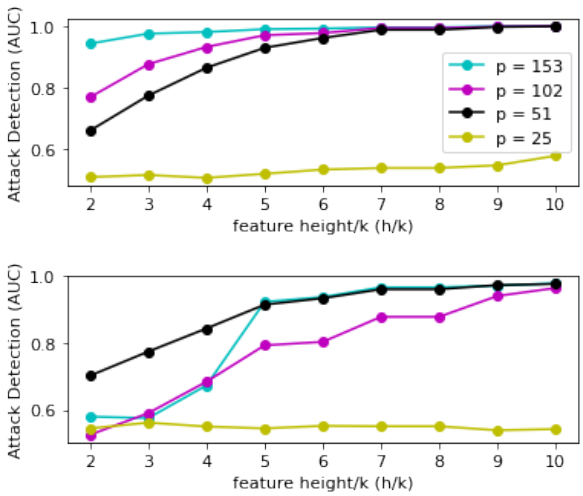

Figure 6: Effect of k in PC module on the AUC of attack detection for FlowNetC on KITTI 2015 (top) and MPI-Sintel (bottom).

7 Conclusion

We propose SADL, the first method to detect and localize patch-based adversarial attacks in optical flow estimators. We use a spatially constrained subset scanning on the inner layer in an unsupervised manner without any training nor prior knowledge of the attacks. We further give insights on which layers are most affected by these attacks for various flow networks. We validated SADL with multiple flow networks and datasets, and compared it with a baseline method, across which SADL consistently performed well. Immediate next step could utilize detected and localized attacks to devise mitigation techniques for flow estimators. Future work also aims to explore various hyper-parameters and further analyze why the performances differ across various networks and datasets.

References

[Agarwal *et al.*, 2009] Sameer Agarwal, Noah Snavely, Ian Simon, Steven Seitz, and Richard Szeliski. Building Rome in a day. *Proceedings of the IEEE International Conference on Computer Vision*, pages 72–79, 2009.

[Akhtar and Mian, 2018] Naveed Akhtar and Ajmal Mian. Threat of adversarial attacks on deep learning in computer vision: A survey. *Ieee Access*, 6:14410–14430, 2018.

[Akinwande *et al.*, 2020] Victor Akinwande, Celia Cintas, Skyler Speakman, and Srihari Sridharan. Identifying audio adversarial examples via anomalous pattern detection. *arXiv preprint arXiv:2002.05463*, 2020.

[Berk and Jones, 1979] Robert H. Berk and Douglas H. Jones. Goodness-of-fit test statistics that dominate the kolmogorov statistics. *Probability Theory and Related Fields*, 47(1):47–59, January 1979.

[Butler *et al.*, 2012] Daniel J. Butler, Jonas Wulff, Garrett B. Stanley, and Michael J. Black. A naturalistic open source movie for optical flow evaluation. In A. Fitzgibbon et al. (Eds.), editor, *Proceedings of the European Conference on Computer Vision*, Part IV, LNCS 7577, pages 611–625. Springer-Verlag, October 2012.

[Chen and Koltun, 2016] Qifeng Chen and Vladlen Koltun. Full flow: Optical flow estimation by global optimization over regular grids. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4706–4714. IEEE Computer Society, 2016.

[Cintas *et al.*, 2020] Celia Cintas, Skyler Speakman, Victor Akinwande, William Ogallo, Komminist Weldemariam, Srihari Sridharan, and Edward McFowland. Detecting adversarial attacks via subset scanning of autoencoder activations and reconstruction error. In Christian Bessiere, editor, *International Joint Conference on Artificial Intelligence*, pages 876–882. International Joint Conferences on Artificial Intelligence Organization, 7 2020. Main track.

[Cintas *et al.*, 2022] Celia Cintas, Payel Das, Girmaw Abebe Tadesse, Brian Quanz, Skyler Speakman, and Pin-Yu Chen. Towards creativity characterization of generative models via group-based subset scanning. In *International Joint Conference on Artificial Intelligence*, pages 4929–4935, 2022.

[Diba *et al.*, 2017] Ali Diba, Mohsen Fayyaz, Vivek Sharma, Amir Hossein Karami, Mohammad Mahdi Arzani, Rahman Yousefzadeh, and Luc Van Gool. Temporal 3d convnets: New architecture and transfer learning for video classification. *CoRR*, abs/1711.08200, 2017.

[Dosovitskiy *et al.*, 2015] Alexey Dosovitskiy, Philipp Fischery, Eddy Ilg, Philip Hausser, Caner Hazirbas, Vladimir Golkov, Patrick van der Smagt, Daniel Cremers, and Thomas Brox. Flownet: Learning optical flow with convolutional networks. In *Proceedings of the IEEE International Conference on Computer Vision*, ICCV '15, pages 2758–2766, Washington, DC, USA, 2015. IEEE Computer Society.

[Geiger *et al.*, 2013] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The kitti dataset. *International Journal of Robotics Research*, 2013.

[Ilg *et al.*, 2017] Eddy Ilg, N. Mayer, Tonmoy Saikia, Margret Keuper, A. Dosovitskiy, and T. Brox. Flownet 2.0: Evolution of optical flow estimation with deep networks. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1647–1655, 2017.

[Kim *et al.*, 2021] Hannah Halin Kim, Shuzhi Yu, and Carlo Tomasi. Joint detection of motion boundaries and occlusions. In *British Machine Vision Conference*, 2021.

[Kim *et al.*, 2022a] Hannah Kim, Girmaw Abebe Tadesse, Celia Cintas, Skyler Speakman, and Kush Varshney. Out-of-distribution detection in dermatology using input perturbation and subset scanning. In *IEEE 19th International Symposium on Biomedical Imaging*, pages 1–4, 2022.

[Kim *et al.*, 2022b] Hannah Halin Kim, Shuzhi Yu, Shuai Yuan, and Carlo Tomasi. Cross-attention transformer for video interpolation. In *Asian Conference on Computer Vision Workshops*, pages 320–337, 2022.

[Le Gall, 1991] Didier Le Gall. Mpeg: A video compression standard for multimedia applications. *Commun. ACM*, 34(4):46–58, April 1991.

[Liu *et al.*, 2019] Pengpeng Liu, Michael Lyu, Irwin King, and Jia Xu. Selflow: Self-supervised learning of optical flow. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4571–4580, 2019.

[McFowland *et al.*, 2013] Edward McFowland, Skyler Speakman, and Daniel B. Neill. Fast generalized subset scan for anomalous pattern detection. *Journal of Machine Learning Research*, 14(12):1533–1561, 2013.

[Menze and Geiger, 2015] Moritz Menze and Andreas Geiger. Object scene flow for autonomous vehicles. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015.

[Neill, 2012] Daniel B. Neill. Fast subset scan for spatial pattern detection. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 74(2):337–360, 2012.

[Papernot *et al.*, 2017] Nicolas Papernot, Patrick McDaniel, Ian Goodfellow, Somesh Jha, Z. Berkay Celik, and Ananthram Swami. Practical black-box attacks against machine learning. In *Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security*, ASIA CCS '17, page 506–519, New York, NY, USA, 2017. Association for Computing Machinery.

[Ranjan *et al.*, 2019] Anurag Ranjan, Joel Janai, Andreas Geiger, and Michael J. Black. Attacking optical flow. *Proceedings of the IEEE International Conference on Computer Vision*, 2019.

[Rhemann *et al.*, 2013] Christoph Rhemann, Asmaa Hosni, Michael Bleyer, Carsten Rother, and Margrit Gelautz. Fast cost-volume filtering for visual correspondence and beyond. *IEEE transactions on Pattern Analysis and Machine Intelligence*, 35(2):504–511, February 2013.

[Schmalfuss *et al.*, 2022] Jenny Schmalfuss, Philipp Scholze, and Andrés Bruhn. A perturbation constrained adversarial attack for evaluating the robustness of optical flow. In *Proceedings of the European Conference on Computer Vision*, 2022.

[Schrodi *et al.*, 2021] Simon Schrodi, Tonmoy Saikia, and Thomas Brox. What causes optical flow networks to be vulnerable to physical adversarial attacks. *CoRR*, abs/2103.16255, 2021.

[Schrodi *et al.*, 2022] Simon Schrodi, Tonmoy Saikia, and Thomas Brox. Towards understanding adversarial robustness of optical flow networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2022.

[Shi and Tomasi, 1994] Jianbo Shi and Carlo Tomasi. Good features to track. In *1994 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 593–600, Seattle, WA, USA, 1994. IEEE Comput. Soc. Press.

[Sun *et al.*, 2018] Deqing Sun, Xiaodong Yang, Ming-Yu Liu, and Jan Kautz. Pwc-net: Cnns for optical flow using pyramid, warping, and cost volume. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[Teed and Deng, 2020] Zachary Teed and Jun Deng. Raft: Recurrent all-pairs field transforms for optical flow. *Proceedings of the European Conference on Computer Vision*, 2020.

[Wortman, 2021] Benjamin Wortman. Hidden patch attacks for optical flow. In *ICML 2021 Workshop on Adversarial Machine Learning*, 2021.

[Xu *et al.*, 2017] Jia Xu, Rene Ranftl, and Vladlen Koltun. Accurate optical flow via direct cost volume processing. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5807–5815, Honolulu, HI, 2017.

[Yamanaka *et al.*, 2021] Koichiro Yamanaka, Keita Takahashi, Toshiaki Fujii, and Ryuraroh Matsumoto. Simultaneous attack on cnn-based monocular depth estimation and optical flow estimation. *IEICE Transactions on Information and Systems*, E104.D(5):785–788, 2021.

[Yu *et al.*, 2022] Shuzhi Yu, Hannah H Kim, Shuai Yuan, and Carlo Tomasi. Unsupervised flow refinement near motion boundaries. In *British Machine Vision Conference*. BMVA Press, 2022.

[Yuan *et al.*, 2022] Shuai Yuan, Xian Sun, Hannah Kim, Shuzhi Yu, and Carlo Tomasi. Optical flow training under limited label budget via active learning. In *Proceedings of the European Conference on Computer Vision*, 2022.

[Yuan *et al.*, 2023] Shuaihang Yuan, Shuzhi Yu, Hannah Kim, and Carlo Tomasi. Semarflow: Injecting semantics into unsupervised optical flow estimation for autonomous driving. *ArXiv*, abs/2303.06209, 2023.