DeSTNet: Densely Fused Spatial Transformer Networks

Roberto Annunziata
roberto.annunziata@onfido.com

Christos Sagonas
christos.sagonas@onfido.com

Jacques Calì
jacques.cali@onfido.com

Onfido Research
3 Finsbury Avenue
London, UK

Abstract

Modern Convolutional Neural Networks (CNN) are extremely powerful on a range of computer vision tasks. However, their performance may degrade when the data is characterised by large intra-class variability caused by spatial transformations. The Spatial Transformer Network (STN) is currently the method of choice for providing CNNs the ability to remove those transformations and improve performance in an end-to-end learning framework. In this paper, we propose *Densely Fused Spatial Transformer Network (DeSTNet)*, which, to our best knowledge, is the first dense fusion pattern for combining multiple STNs. Specifically, we show how changing the connectivity pattern of multiple STNs from *sequential* to *dense* leads to more powerful alignment modules. Extensive experiments on three benchmarks namely, MNIST, GTSRB, and IDocDB show that the proposed technique outperforms related state-of-the-art methods (i.e., STNs and CSTNs) both in terms of accuracy and robustness.

1 Introduction

Recently, significant progress has been made in several real-world computer vision applications, including image classification [13, 22], face recognition [32], object detection and semantic segmentation [12, 14, 31]. These breakthroughs are attributed to advances of CNNs [13, 16, 33], as well as the availability of huge amounts of data [21, 22] and computational power. In general, performance is adversely affected by intra-class variability caused by spatial transformations, such as affine or perspective; therefore, achieving invariance to the aforementioned transformations is highly desirable. CNNs achieve translation equivariance through the use of convolutional layers. However, the filter response is not in itself transformation invariant. To compensate for this max-pooling strategies are often applied [4, 22]. Pooling is usually performed on very small regions (e.g., 2×2), giving it an effective rate of only a few pixels, increasing as we go deeper. Another technique used to achieve invariance is data augmentation [22]. Specifically, a set of known transformations are applied to training samples. However, this approach has the following disadvantages: *(i)* the set of transformations must be defined *a-priori*; and *(ii)* a large number of samples are required, thus reducing training efficiency.
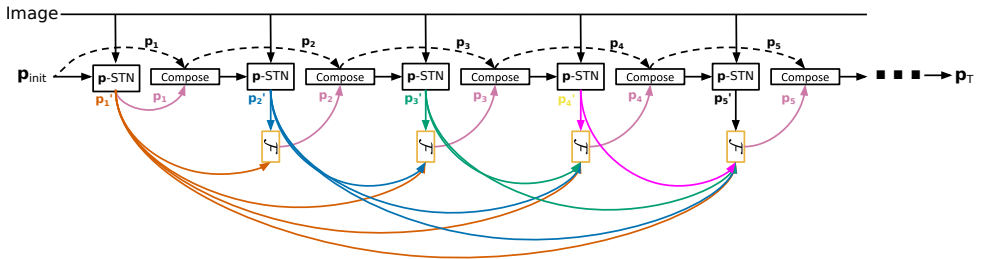
Figure 1: DeSTNet - A stack of Densely fused Spatial Transformer Networks.

Arguably, one of the best known methods used to efficiently increase invariance to geometric transformations in CNNs is the Spatial Transformer Network (STN) [19]. STN provides an end-to-end learning mechanism that can be seamlessly incorporated into a CNN to explicitly learn how to transform the input data to achieve spatial invariance. One might look at an STN as an attention mechanism that manipulates a feature map in a way that the input is simplified for some process downstream, e.g. image classification. For example, in [5] an STN was used in a supervised manner in order to improve the performance of a face detector. Similarly, a method based on STN for performing simultaneously face alignment and recognition was introduced in [33]. Although the incorporation of the STN within CNNs led to state-of-the-art performance, its effectiveness could reduce drastically in cases where the face is heavily deformed (e.g. due to facial expressions). To overcome this issue, Wu *et al.* [37] proposed multiple STNs linked in a recurrent manner. One of the main drawbacks when combining multiple STNs can be seen in the boundary pixels. Each STN samples the output image produced by the previous, thus as the image passes through multiple transforms the quality of the transformed image deteriorates. In cases where initial bounding boxes are not of sufficient accuracy, transformed images are heavily affected by the boundary effect, shown in [25]. To overcome this and inspired by the Lucas-Kanade algorithm [27], Lin and Lucey [25] proposed Compositional STNs (CSTNs) and their recurrent version ICSTNs. CSTNs are made up of an STN variant (henceforth, **p**-STN), which propagates transformation parameters instead of the transformed images.

In this work, building on the success of **p**-STNs, we present DeSTNet (Fig. 1), an end-to-end framework designed to increase spatial invariance in CNNs. Firstly, motivated by information theory principles, we propose a dense fusion connectivity pattern for **p**-STNs. Secondly, we introduce a novel *expansion-contraction* fusion block for combining the predictions of multiple **p**-STNs in a *dense* manner. Finally, extensive experimental results on two public benchmarks and a non-public real-world dataset suggest that the proposed DeSTNet outperforms the state-of-the-art CSTN[25] and the original STN [19].

2 Related Work

Geometric transformations can be mitigated through the use of either *(i)* invariant or equivariant features; *(ii)* encoding some form of attention mechanism. More traditional computer vision systems achieved this through the use of hand-crafted features such as HOG [9], SIFT [26] and SCIRD [1, 2] that were designed to be invariant to various transformations. In CNNs translation equivariance is achieved through convolutions and limited spatial invariance from pooling.

In [20], a method for creating scale-invariant CNNs was proposed. Locally scale-invariant representations are obtained by applying filters at multiple scales and locations followed by max-pooling. Rotational invariance can be achieved by discretely rotating the filters [6, 7, 28] or input images and feature maps [10, 23, 30]. Recently, a method for providing continuous rotation robustness was proposed in [36]. To facilitate the translation invariance property of CNNs, Henriques and Vedaldi [13] proposed to transform the image via a constant warp and then employ a simple convolution. Although, the aforementioned is very simple and powerful, it requires prior knowledge of the type of transformation as well as the location inside the image where it is applied.

More related to our work are methods that encode an attention or detection mechanism. Szegedy *et al*. [35] introduced a detection system as a form of regression within the network to predict object bounding boxes and classification results simultaneously. Erhan *et al*. [11] proposed a saliency-inspired neural network that predicts a set of class-agnostic bounding boxes along with a likelihood of each box containing the object of interest. A few years later, He *et al*. [14] designed a network that performs a number of complementary tasks: classification, bounding box prediction and object segmentation. The region proposal network within their model provided a form of learnt attention mechanism. For a more thorough review of object detection systems we point the reader to Huang *et al*. [17] who look at speed/accuracy trade-offs for modern detection systems.

3 Methodology

Let $\mathcal{D} = \{\mathbf{I}_1, \mathbf{I}_2, \dots, \mathbf{I}_M\}$ be a set of M images and $\{\mathbf{p}_i\}_{i=1}^M \in \mathbb{R}^n$ ($n = 8$ for perspective) the initial estimates[1] of the distortion parameters for each image. Our goal is to reduce the intra-class variability due to the perspective transformations inherently applied to the images during capture. Achieving this goal has the potential to significantly simplify subsequent tasks, such as classification. To this end, we need to find the optimal parameters $\{\mathbf{p}_i^*\}_{i=1}^M$ that warp all the images into a transformation-free space.

Arguably, the most notable method for finding the optimal parameters is the STN [19]. An STN is made up of three components, namely the *localization network*, the *grid generator* and the *sampler*. The *localization network* $\mathbf{L}$ is used to predict transformation parameters for a given input image $\mathbf{I}$ and initial parameters $\mathbf{p}_{\text{init}}$, i.e. $\mathbf{p} = \mathbf{L}(\mathbf{I}, \mathbf{p}_{\text{init}})$, the *grid generator* and *sampler* are used for warping the image based on the computed parameters, i.e. $\mathbf{I}(\mathcal{W}(\mathbf{p}))$ (Fig. 2(a)). By allowing the network to learn how to warp the input, it is able to gain geometric invariance, thus boosting task performance. When recovering larger transformations a number of STNs can be stacked or used in combination with a recurrent framework (Fig. 2(b)). However, this tends to introduce boundary artifacts and image quality degradation in the final transformed image, as each STN re-samples from an image that is the result of multiple warpings.

To address the aforementioned and inspired by the success of the LK algorithm for image alignment, Lin and Lucey [25] proposed compositional STNs (CSTNs). The LK algorithm is commonly used for alignment problems [3, 29] as it approximates the linear relationship between appearance and geometric displacement. Specifically, given two images $\mathbf{I}_1, \mathbf{I}_2$ that are related by a parametric transformation $\mathcal{W}$, the goal of LK is to find the optimal parameters that minimize the ℓ_2 norm of the error between the deformed version of $\mathbf{I}_1$, and $\mathbf{I}_2$:

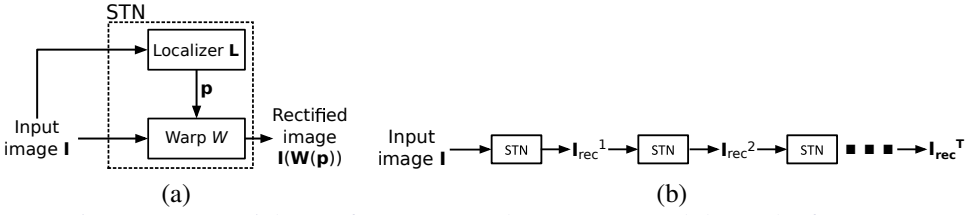[1]This initial estimates may simply be identities.

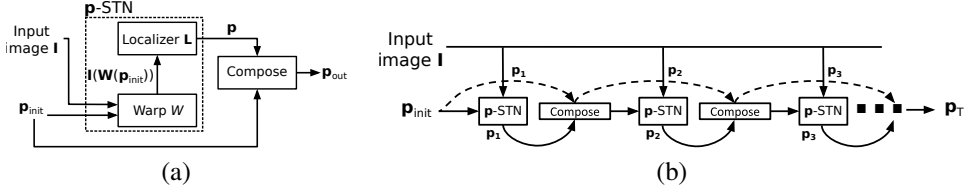Figure 2: (a) Spatial Transformer Network (STN) [19] and (b) stack of STNs.

Figure 3: (a) Compositional STN (CSTN) [25] and (b) stack of CSTNs.

$\min_{\mathbf{p}} \|\mathbf{I}_1(\mathcal{W}(\mathbf{p})) - \mathbf{I}_2\|_2^2$. Applying first-order Taylor expansion to $\mathbf{I}_1$, it has been shown that the previous problem can be optimised by an iterative algorithm with the following additive-based update rule:

$$\mathbf{p}_{t+1} = \mathbf{p}_t + \Delta \mathbf{p}_t, \tag{1}$$

at each iteration t. In [25], Lin and Lucey introduced the CSTN that predicts the parameters' updates by employing a modified STN, which we refer to as **p**-STN, and then compose them as in Eq. (1). By incorporating the LK formulation, the resulting CSTN is able to inherit the geometry preserving property of LK. Unlike a stack of STNs that propagates *warped images* to recover large displacements (Fig. 2(b)), a stack of CSTNs (Fig. 3(b)) propagate the *warp parameters* in a similar fashion to the iterative process used in the LK algorithm.

Here, we extend the CSTN framework to improve the information flow in terms of parameters' updates. In particular, we modify Eq. (1) and propose the additive-based *dense fusion* update rule:

$$\mathbf{p}_{t+1} = \mathbf{p}_t + f(\Delta \mathbf{p}'_t, \Delta \mathbf{p}'_{t-1}, \dots, \Delta \mathbf{p}'_1), \tag{2}$$

where the parameters' update at iteration t, $\Delta \mathbf{p}_t$, is now a function $f : \mathbb{R}^{n \times t} \to \mathbb{R}^n$ of the updates predicted by the **p**-STN at iteration t, $\Delta \mathbf{p}'_t$, and *all* the previous ones, $\{\Delta \mathbf{p}'_i\}_{i=1}^{t-1}$ (Fig. 1). Learning the *fusion* function $f(\cdot)$ at each iteration t means learning the posterior distribution $p(\Delta \mathbf{p}_t | \Delta \mathbf{p}'_t, \Delta \mathbf{p}'_{t-1}, \dots, \Delta \mathbf{p}'_1)$ for the parameters' update $\Delta \mathbf{p}_t$. From an information theory perspective, this amounts to predicting $\Delta \mathbf{p}_t$ with an *uncertainty* measured by the conditional entropy, $\mathcal{H}(\Delta \mathbf{p}_t | \Delta \mathbf{p}'_t, \Delta \mathbf{p}'_{t-1}, \dots, \Delta \mathbf{p}'_1)$. We notice that the CSTN update in Eq. (1) is a special case of Eq. (2):

$$\mathbf{p}_{t+1} = \mathbf{p}_t + f(\Delta \mathbf{p}'_t), \tag{3}$$

where the parameters' update at iteration t, $\Delta \mathbf{p}_t$, is *only* a function of the update predicted by the tth regressor ($\Delta \mathbf{p}'_t$). In fact, no fusion has to be applied (i.e., $f(\cdot)$ is an identity mapping) and $\Delta \mathbf{p}_t = \Delta \mathbf{p}'_t$. In other words, the CSTN learns the distribution $p(\Delta \mathbf{p}_t)$ for the parameters' update $\Delta \mathbf{p}_t$ at each iteration t. This amounts to predicting $\Delta \mathbf{p}_t$ with an *uncertainty* measured by the related entropy, $\mathcal{H}(\Delta \mathbf{p}_t)$. Invoking the well-known *'conditioning reduces entropy'* principle from information theory [8], it can be shown that $\mathcal{H}(\Delta \mathbf{p}_t | \Delta \mathbf{p}'_t, \Delta \mathbf{p}'_{t-1}, \dots, \Delta \mathbf{p}'_1) \leq$
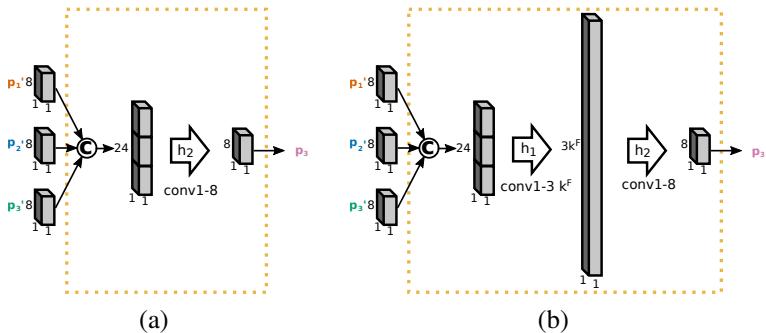
Figure 4: Fusion blocks. (a) The bottleneck-based fusion block proposed in [16]. (b) The proposed *expansion-contraction* fusion block used in Figure 1.

$\mathcal{H}(\Delta \mathbf{p}_t)$. In other words, the update predictions in the proposed formulation are *upper-bounded* by those made with CSTN in terms of uncertainty. We advocate that this theoretical advantage can translate into better performance.

Inspired by the recent success of densely connected CNNs [16] and justified by the extension outlined above, we propose an alignment module which we call *DeSTNet* (Densely fused Spatial Transformer Network). DeSTNet consists of a cascade of **p**-STNs with a dense fusion connectivity pattern, as shown in Fig. 1. The fusion function, implemented by the fusion block $\mathcal{F}$ in Fig. 1, is adopted to combine the update predictions of *all* the previous **p**-STNs and estimate the best parameters' update at each level t. Unlike the fusion blocks adopted in [16] consisting of a single bottleneck layer (Fig. 4(a)), we advocate the use of an *expansion-contraction* fusion block (Fig. 4(b)). This solves the fusion task in a high-dimensional space and then maps the result back to the original. Specifically, we concatenate all the previous parameters' updates and project them using a 1×1 convolution layer with depth $n \times t \times k^F$ (*expansion*), where n is the dimension of the warp parameters **p**, $t = 1, \ldots, T$ is the level within DeSTNet, and k^F is the *expansion rate*. This is then followed by a $1 \times 1 \times n$ convolution layer (*contraction*), as shown in Fig. 4(b). We adopt tanh activations (non-linearities) after each convolutional layer of the fusion block to be able to predict both positive and negative parameter values. It is worth noting that the use of expansion layers is made possible by the relatively low dimension of each individual prediction (i.e., $n = 8$ for perspective warps).

4 Experiments

In this section, we assess the effectiveness of the proposed DeSTNet in *(i)* adding spatial transformation invariance (up to perspective warps) to CNN-based classification models and *(ii)* planar image alignment. To this end, artificially distorted versions of two widely used datasets, namely the German Traffic Sign Recognition Benchmark (GTSRB) [34] and MNIST [24] are utilised. Furthermore, we evaluate the performance of DeSTNet on a non-public dataset of official identity documents (IDocDB), which includes substantially larger images (e.g. up to $6,016 \times 3,910$ pixels) and, more importantly, real perspective transformations. Additional results can be found in supplementary material.

	Model	Test Error	Architecture	
			Alignment	Classifier
GTSRB	CNN	8.29%	conv7-6 | conv7-12 | P | conv7-24 | FC(200) | FC(43)	
	STN	6.49%	conv7-6 | conv7-24 | FC(8)	conv7-6 | conv7-12 | P | FC(43)
	CSTN-1	5.01%	[conv7-6 | conv7-24 | FC(8)]×1	conv7-6 | conv7-12 | P | FC(43)
	ICSTN-4	3.18%	[conv7-6 | conv7-24 | FC(8)]×4	conv7-6 | conv7-12 | P | FC(43)
	CSTN-4	3.15%	[conv7-6 | conv7-24 | FC(8)]×4	conv7-6 | conv7-12 | P | FC(43)
	DeSTNet-4	**1.99%**	$\mathcal{F}$\{[conv7-6 | conv7-24 | FC(8)]×4\}	conv7-6 | conv7-12 | P | FC(43)
MNIST	CNN	6.60%	conv3-3 | conv3-6 | P | conv3-9 | conv3-12 | FC(48) | FC(10)	
	STN	4.94%	conv7-4 | conv7-8 | P | FC(48) | FC(8)	conv9-3 | FC(10)
	CSTN-1	3.69%	[conv7-4 | conv7-8 | P | FC(48) | FC(8)]×1	conv9-3 | FC(10)
	ICSTN-4	1.23%	[conv7-4 | conv7-8 | P | FC(48) | FC(8)]×4	conv9-3 | FC(10)
	CSTN-4	1.04%	[conv7-4 | conv7-8 | P | FC(48) | FC(8)]×4	conv9-3 | FC(10)
	DeSTNet-4	**0.71%**	$\mathcal{F}$\{[conv7-4 | conv7-8 | P | FC(48) | FC(8)]×4\}	conv9-3 | FC(10)

Table 1: Test classification errors of the compared models on GTSRB and MNIST datasets.

4.1 Image Classification

Traffic Signs: We report experimental results on the GTSRB dataset [34], consisting of 39,209 training and 12,630 test colour images from 43 traffic signs taken under various real-world conditions including motion blur, illumination changes and extremely low resolution. We adopt the image classification error as a *proxy* measure for alignment quality. Specifically, we build classification pipelines made up of two components: an alignment network followed by a classification one (detailed architectures reported in Table 1). Both networks are jointly trained with the classification-based loss using standard back-propagation. At parity of a classification network, a lower classification error suggests better alignment (i.e., spatial transformation invariance). Following the experimental protocol in [25], we resize images to $s \times s$, $s = 36$ pixels and artificially distort them using a perspective warp. Specifically, the four corners of each image are independently and randomly scaled with Gaussian noise $\mathcal{N}(0, (\sigma s)^2)$, then randomly translated with the same noise model.

In the first experiment, we follow the same setting adopted in [25] and train all the networks for 200,000 iterations with a batch of 100 perturbed samples generated on the fly. For DeSTNet, we use $\alpha_{clf} = 10^{-2}$ as the learning rate for the classification network and $\alpha_{aln} = 10^{-4}$ for the alignment network which is reduced by 10 after 100,000 iterations. For the proposed *expansion-contraction* fusion block we set the expansion rate $k^F = 256$, as a good trade-off between speed and performance, we use dropout with keep probability equal to $S = 0.9$. Finally, images of both train and test sets are perturbed using $\sigma = 10\%$, corresponding to a maximum perturbation of 3.6 pixels.

We compare the performance of DeSTNet to the most related methods, STN [19], a single CSTN (CSTN-1) [25], and stack of four CSTNs (CSTN-4) [25]. For completeness, we report classification results of a CNN with roughly the same number of learnable parameters and the recurrent version of CSTN (i.e., ICSTN) [25]. To isolate the contribution of the alignment module, we adopt the same CNN classifier for all. By examining Table 1[2] we observe that alignment improves classification performance, irrespective of the specific alignment module, supporting the need for removing perspective transformations with which a standard CNN classifier would not be able to cope.[3] Importantly, CSTN-1 achieves lower classification error as compared to the STN (5.01% vs 6.49%), thus supporting our architectural

---

[2] convD$_1$-D$_2$: convolution layer with D$_1$ × D$_1$ receptive field and D$_2$ channels, P: max-pooling layer, FC: fully connected layer, $\mathcal{F}$: fusion operation used in DeSTNet for combining the parameters' updates, $\tilde{\mathcal{F}}$: standard fusion operation [16].

[3] Convolution and max-pooling help with small transformations, but are not enough to cope with full perspective warpings.

	Model	Test error			Architecture	
		Perturbation σ			Alignment	Classifier
		10%	20%	30%		
GTSRB	CSTN-4	6.86%	8.92%	13.72%	[conv7-6	conv7-24	FC(8)]×4	FC(43)
	DeSTNet-4 ($\tilde{\mathcal{F}}$)	3.60%	4.65%	5.25%	$\tilde{\mathcal{F}}$\{[conv7-6	conv7-24	FC(8)]×4\}	FC(43)
	DeSTNet-4	**3.04%**	**3.80%**	**3.85%**	$\mathcal{F}$\{[conv7-6	conv7-24	FC(8)]×4\}	FC(43)
MNIST	C-STN-4	1.50%	2.39%	3.40%	[conv7-4	conv7-8	P	FC(48)	FC(8)]×4	FC(10)
	DeSTNet-4 ($\tilde{\mathcal{F}}$)	0.86%	0.89%	1.09%	$\tilde{\mathcal{F}}$\{[conv7-4	conv7-8	P	FC(48)	FC(8)]×4\}	FC(10)
	DeSTNet-4	**0.66%**	**0.72%**	**0.74%**	$\mathcal{F}$\{[conv7-4	conv7-8	P	FC(48)	FC(8)]×4\}	FC(10)

Table 2: Test classification errors of the compared models by using a single fully connected layer as classifier under three perturbation levels on GTSRB and MNIST datasets.

choice of building DeSTNet using **p**-STNs. Moreover, using a cascade of four CSTNs further improves results. Finally, the DeSTNet-4 outperforms CSTN-4 with an error of 1.99% down from 3.15% which amounts to a relative improvement of 37%.

It is worth noting, *(i)* the perturbations in this experiment are relatively small ($\sigma = 10\%$) and *(ii)* the CNN network followed by a fully connected layer as classifier does not fully offload the alignment task to the alignment network. This is due to the translation invariance and robustness to small transformations brought about by the convolutions and pooling layers. Therefore, to further investigate the alignment quality of the state-of-the-art CSTN and DeSTNet, we use a single fully connected layer as a classification network and report performance under three perturbation levels $\sigma = \{10\%, 20\%, 30\%\}$ corresponding to a minimum of 3.6 and a maximum of 10.8 pixels. Results in Table 2[2] show that, *(i)* DeSTNet yields an alignment quality that significantly simplifies the classification task compared to CSTN (i.e., up to 9.87% better classification performance for DeSTNet); *(ii)* DeSTNet exhibits robustness against stronger perturbation levels, with performance degrading by only 0.81% from 10% to 30% perturbation, while CSTN performance degrades by 6.86% in the same range; and *(iii)* the proposed *expansion-contraction* fusion block $\mathcal{F}$ leads to better performance w.r.t. the standard bottleneck layer $\tilde{\mathcal{F}}$ proposed in [16]. Qualitative experimental results for CSTN and DeSTNet under different perturbation levels are reported in Fig. 5. More

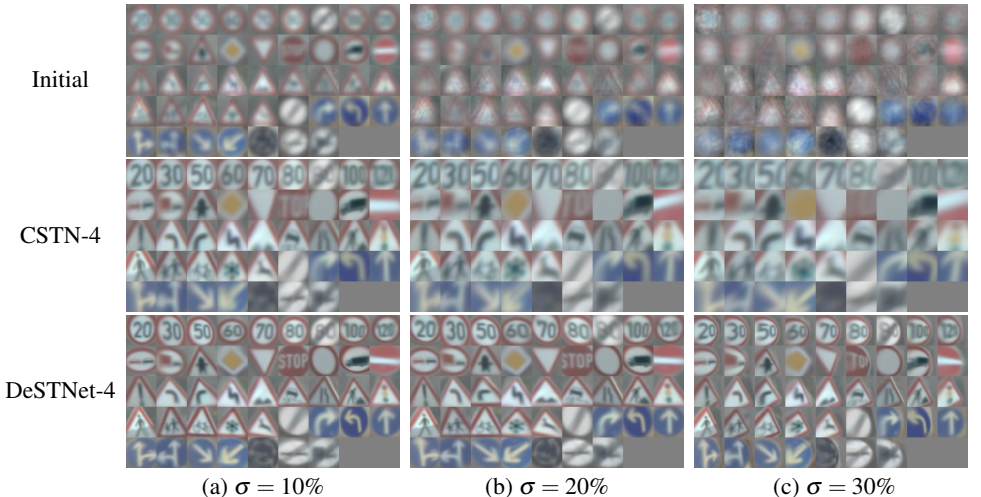(a) $\sigma = 10\%$　　　　(b) $\sigma = 20\%$　　　　(c) $\sigma = 30\%$

Figure 5: Qualitative comparison of CSTN-4 and DeSTNet-4 methods on GTSRB dataset. Averages of the test traffic signs under different perturbation levels.
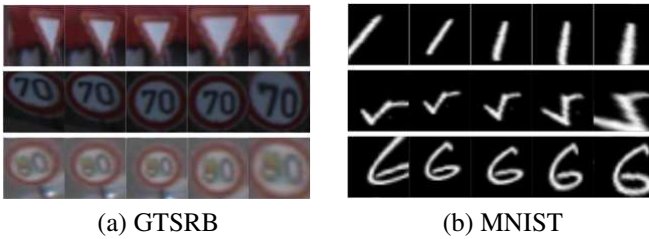
(a) GTSRB   (b) MNIST

Figure 6: Sample alignment results produced by the DeSTNet-4 model on three examples (rows) from the GTSRB (a) and the MNIST (b) datasets. Column 1: input image; columns 2-5: results obtained by applying the intermediate perspective transformations predicted at levels 1-4, respectively.

specifically, the averages of the 43 traffic signs before and after convergence for CSTN-4 and DeSTNet-4 are shown. We observe that the average images produced by DeSTNet-4 are much sharper and have more details (even for the 30% perturbation level, Fig. 5(c)) than the averages produced by CSTN-4, this is indicative of the better alignment performance for the proposed model. Fig. 6(a) illustrates aligned examples generated by DeSTNet-4.

Handwritten Digits: For this experiment, we adopt MNIST dataset [24], consisting of handwritten digits between 0 and 9, with a training set of $60,000$ and $10,000$ test grayscale images (28×28 pixels). We adopt the same settings as for the GTSRB experiments by using the image classification error as a *proxy* measure for alignment quality. Training and test sets are distorted using the same perspective warp noise model ($\sigma = 12.5\%$, corresponding to a maximum perturbation of 3.5 pixels).

Experimental results are reported in Table 1^2. In line with the GTSRB experiments, *(i)* pre-alignment considerably improves classification performance, regardless of the specific alignment module used; *(ii)* lower classification error is achieved when using CSTN-1 as compared to STN, again supporting our choice of using **p**-STNs as base STNs in DeST-Net; *(iii)* although performance almost saturates with four CSTNs, DeSTNet is still able to squeeze extra performance, outperforming CSTN-4 with an error of 0.71% down from 1.04% which is a relative improvement of 32%.

We further investigate the alignment quality of the state-of-the-art CSTN and DeSTNet, when a single fully connected layer is used for classification and report performance under three perturbation levels corresponding to a minimum of 2.8 pixels and a maximum of 8.4 pixels. By inspecting the results reported in Table 2^2, we can see that, *(i)* DeSTNet achieves an alignment quality that significantly simplifies the classification task compared to CSTN (i.e., up to 2.66% better classification performance for DeSTNet); *(ii)* DeSTNet exhibits robustness against stronger perturbation levels, with the classification performance degrading

(a) $\sigma = 10\%$ (b) $\sigma = 20\%$ (c) $\sigma = 30\%$
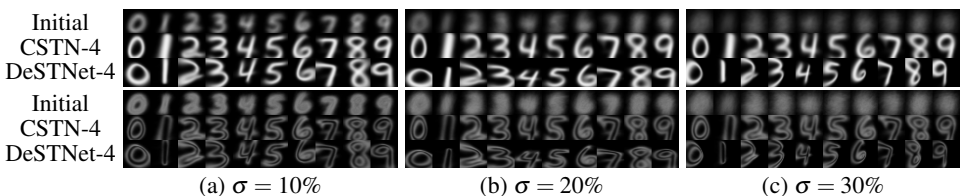
Figure 7: Qualitative comparison of CSTN-4 and DeSTNet-4 on the MNIST dataset. Mean (top rows) and variance (bottom rows) of the 10 digits under different perturbation levels.
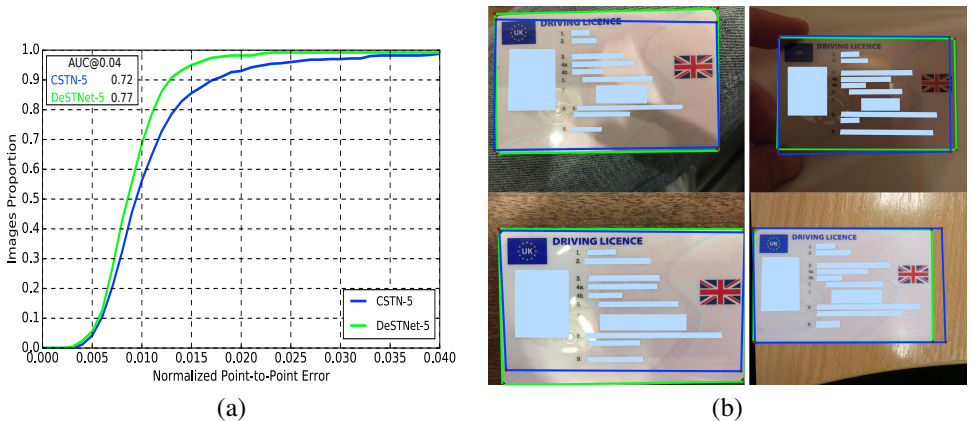
Figure 8: (a) Cumulative Error Distribution curves and (b) qualitative results obtained by the **CSTN-5** and **DeSTNet-5** on IDocDB.

by only 0.08% from 10% to 30% perturbation, while CSTN performance degrades by 1.90% in the same range; and *(iii)* the proposed *expansion-contraction* fusion block further helps reducing the classification test error.

Qualitative experimental results are reported in Fig. 7. In particular, the average and corresponding variance of all test samples grouped by digit are computed and shown for CSTN-4 and DeSTNet-4. Inspecting the images we can see that the mean images generated by DeSTNet-4 are sharper than those of CSTN-4 while the variance ones are thinner. This suggests that DeSTNet is more accurate and robust to different perturbation levels compared to CSTN. Finally, aligned images generated by the DeSTNet-4 are displayed in Fig. 6(b).

4.2 Document Alignment

Here, we show how DeSTNet can be successfully utilised for aligning planar images. To this end, we make use of our non-public official identity documents dataset (IDocDB) consisting of $1,000$ training and 500 testing colour images collected under in-the-wild conditions. Specifically, each image contains a single identity document (UK Driving Licence V2015) and their size ranges from 422×215 to $6,016 \times 3,910$ pixels. In addition to typical challenges such as non-uniform illumination, shadows, and compression noise, several other aspects make this dataset challenging, including: the considerable variations in resolution; highly variable background which may include clutter and non-target objects; occlusion, e.g. the presence of fingers covering part of the document when held for capture. The ground truth consists of the location of the four corners of each document. From these points, we can compute a homography matrix that maps each document to a reference frame. The alignment task can be solved by predicting the location of the corner points on each input image. We train the networks using the smooth ℓ_1 loss [51] between the ground truth and the predicted corner coordinates.

Adopting the following experimental setting: we resize each image to 256×256 pixels for computational efficiency, as done for instance in [18, 53]. We set the learning rate for the localisation network to $\alpha_{\text{aln}} = 10^{-4}$, which we reduce by 10 after $20,000$ iterations. We use batches with 8 images each for all the models. For the fusion blocks of DeSTNet, we set $k^F = 256$ and use $S = 0.9$. We assess the performance of DeSTNet and compare it with

the state-of-the-art CSTNs (strongest baseline based on the presented experiments). Given the increased complexity of the task compared to MNIST and GTSRB, we built networks with five STNs for both CSTN and DeSTNet (architectures are reported in Table 1 of supplementary material). For comparison, we use the average point-to-point Euclidean distance, normalised by each document's diagonal, between the ground truth and predicted location of the four corners. In addition, the Cumulative Error Distribution (CED) curve for each method is computed using the fraction of test images for which the average error is smaller than a threshold. The CED curves in Fig. 8(a) show that DeSTNet-5 outperforms CSTN-5 both in terms of accuracy and robustness. In fact, DeSTNet achieves a higher AUC@0.04 (0.77 vs 0.72). Qualitative results for CSTN and DeSTNet are displayed in Fig. 8(b).

5   Conclusions

It is well-known that image recognition is adversely affected by spatial transformations. Increasing geometric invariance helps to improve performance. Although CNNs achieve some level of translation equivariance, they are still susceptible to large spatial transformations. In this paper, we address this problem by introducing DeSTNet, a stack of densely fused STNs that improve information flow in terms of warp parameters' updates. Furthermore, we provide a novel fusion technique demonstrating its improved performance in our problem setting. We show the superiority of DeSTNet over the current state-of-the-art STN and its variant CSTN, by conducting extensive experiments on two widely-used benchmarks (MNIST, GTSRB) and a new non-public real-world dataset of official identity documents.

References

[1] Roberto Annunziata and Emanuele Trucco. Accelerating convolutional sparse coding for curvilinear structures segmentation by refining SCIRD-TS filter banks. *IEEE Transactions on Medical Imaging (IEEE–TMI)*, 35(11):2381–2392, 2016.

[2] Roberto Annunziata, Ahmad Kheirkhah, Pedram Hamrah, and Emanuele Trucco. Scale and curvature invariant ridge detector for tortuous and fragmented structures. In *International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, pages 588–595. Springer, 2015.

[3] Simon Baker and Iain Matthews. Lucas-kanade 20 years on: A unifying framework. *International Journal of Computer Vision (IJCV)*, 56(3):221–255, 2004.

[4] Y-Lan Boureau, Jean Ponce, and Yann LeCun. A theoretical analysis of feature pooling in visual recognition. In *Proceedings of International Conference on Machine Learning (ICML)*, pages 111–118, 2010.

[5] Dong Chen, Gang Hua, Fang Wen, and Jian Sun. Supervised transformer network for efficient face detection. In *Proceedings of European Conference on Computer Vision (ECCV)*, pages 122–138. Springer, 2016.

[6] Taco Cohen and Max Welling. Group equivariant convolutional networks. In *Proceedings of International Conference on Machine Learning (ICML)*, pages 2990–2999, 2016.

[7] Taco S Cohen and Max Welling. Steerable cnns. *Proceedings of International Conference on Learning Representations (ICLR)*, 2017.

[8] Thomas M Cover and Joy A Thomas. *Elements of information theory*. John Wiley & Sons, 2012.

[9] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, volume 1, pages 886–893, 2005.

[10] Sander Dieleman, Jeffrey De Fauw, and Koray Kavukcuoglu. Exploiting cyclic symmetry in convolutional neural networks. In *Proceedings of International Conference on Machine Learning (ICML)*, 2016.

[11] Dumitru Erhan, Christian Szegedy, Alexander Toshev, and Dragomir Anguelov. Scalable object detection using deep neural networks. In *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pages 2147–2154, 2014.

[12] Ross Girshick. Fast r-cnn. In *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pages 1440–1448, 2015.

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pages 770–778, 2016.

[14] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, pages 2980–2988, 2017.

[15] Joao F Henriques and Andrea Vedaldi. Warped convolutions: Efficient invariance to spatial transformations. In *Proceedings of International Conference on Machine Learning (ICML)*, 2017.

[16] Gao Huang, Zhuang Liu, Kilian Q Weinberger, and Laurens van der Maaten. Densely connected convolutional networks. In *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, volume 1, page 3, 2017.

[17] Jonathan Huang, Vivek Rathod, Chen Sun, Menglong Zhu, Anoop Korattikara, Alireza Fathi, Ian Fischer, Zbigniew Wojna, Yang Song, Sergio Guadarrama, et al. Speed/accuracy trade-offs for modern convolutional object detectors. In *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, 2017.

[18] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pages 1125–1134, 2017.

[19] Max Jaderberg, Karen Simonyan, Andrew Zisserman, et al. Spatial transformer networks. In *Proceedings of Advances in Neural Information Processing Systems (NIPS)*, pages 2017–2025, 2015.

[20] Angjoo Kanazawa, Abhishek Sharma, and David Jacobs. Locally scale-invariant convolutional neural networks. *arXiv preprint arXiv:1412.5104*, 2014.

[21] Ira Kemelmacher-Shlizerman, Steven M Seitz, Daniel Miller, and Evan Brossard. The megaface benchmark: 1 million faces for recognition at scale. In *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pages 4873–4882, 2016.

[22] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Proceedings of Advances in Neural Information Processing Systems (NIPS)*, pages 1097–1105, 2012.

[23] Dmitry Laptev, Nikolay Savinov, Joachim M Buhmann, and Marc Pollefeys. Ti-pooling: transformation-invariant pooling for feature learning in convolutional neural networks. In *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pages 289–297, 2016.

[24] Yann LeCun. The mnist database of handwritten digits. *http://yann. lecun. com/exdb/mnist/*, 1998.

[25] Chen-Hsuan Lin and Simon Lucey. Inverse compositional spatial transformer networks. In *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pages 2568–2576, 2017.

[26] David G Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision (IJCV)*, 60(2):91–110, 2004.

[27] Bruce D Lucas and Takeo Kanade. An iterative image registration technique with an application to stereo vision. In *Proceedings of International Joint Conference on Artificial Intelligence (IJCAI)*, pages 674–679, 1981.

[28] Diego Marcos, Michele Volpi, and Devis Tuia. Learning rotation invariant convolutional filters for texture classification. In *Proceedings of International Conference on Pattern Recognition (ICPR)*, pages 2012–2017, 2016.

[29] Iain Matthews and Simon Baker. Active appearance models revisited. *International Journal of Computer Vision (IJCV)*, 60(2):135–164, 2004.

[30] Edouard Oyallon and Stéphane Mallat. Deep roto-translation scattering for object classification. In *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, volume 3, page 6, 2015.

[31] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Proceedings of Advances in Neural Information Processing Systems (NIPS)*, pages 91–99, 2015.

[32] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pages 815–823, 2015.

[33] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *Proceedings of International Conference on Learning Representations (ICLR)*, 2014.

[34] Johannes Stallkamp, Marc Schlipsing, Jan Salmen, and Christian Igel. The german traffic sign recognition benchmark: a multi-class classification competition. In *Proceedings of International Joint Conference on Neural Networks (IJCNN)*, pages 1453–1460, 2011.

[35] Christian Szegedy, Alexander Toshev, and Dumitru Erhan. Deep neural networks for object detection. In *Proceedings of Advances in Neural Information Processing Systems (NIPS)*, pages 2553–2561, 2013.

[36] Daniel E Worrall, Stephan J Garbin, Daniyar Turmukhambetov, and Gabriel J Brostow. Harmonic networks: Deep translation and rotation equivariance. In *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, volume 2, 2017.

[37] Wanglong Wu, Meina Kan, Xin Liu, Yi Yang, Shiguang Shan, and Xilin Chen. Recursive spatial transformer (rest) for alignment-free face recognition. In *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, pages 3772–3780, 2017.

[38] Yuanyi Zhong, Jiansheng Chen, and Bo Huang. Toward end-to-end face recognition through alignment learning. *IEEE Signal Processing Letters*, 24(8):1213–1217, 2017.