PRISE: Demystifying Deep Lucas-Kanade with Strongly Star-Convex Constraints for Multimodel Image Alignment

Yiqing Zhang Xinming Huang Ziming Zhang
Worcester Polytechnic Institute
100 Institute Rd, Worcester, MA, USA
{yzhang37, xhuang, zzhang15}@wpi.edu

Abstract

The Lucas-Kanade (LK) method is a classic iterative homography estimation algorithm for image alignment, but often suffers from poor local optimality especially when image pairs have large distortions. To address this challenge, in this paper we propose a novel Deep Star-Convexified Lucas-Kanade (PRISE) *method for multimodel image alignment by introducing strongly star-convex constraints into the optimization problem. Our basic idea is to enforce the neural network to approximately learn a star-convex loss landscape around the ground truth give any data to facilitate the convergence of the LK method to the ground truth through the high dimensional space defined by the network. This leads to a minimax learning problem, with contrastive (hinge) losses due to the definition of strong star-convexity that are appended to the original loss for training. We also provide an efficient sampling based algorithm to leverage the training cost, as well as some analysis on the quality of the solutions from PRISE. We further evaluate our approach on benchmark datasets such as MSCOCO, GoogleEarth, and GoogleMap, and demonstrate state-of-the-art results, especially for small pixel errors. Code can be downloaded from* `https://github.com/Zhang-VISLab`.

1. Introduction

Deep learning networks have achieved great success in homography estimation by directly predicting the transformation matrix in various scenarios. However, the existing classic algorithms still take the place for showing more explainability compared with the deep learning architectures. Such algorithms are often rooted from well-studied theoretical and empirical grounding. Current works often focus on combining the robustness of deep learning with explainability of classical algorithms to handle multimodel image alignment such as image modality and satellite modality. However, due to the high nonconvexity in homography estimation, such methods often suffer from poor local optimality.

Recently Zhao *et al.* [77] proposed DeepLK for multimodel image alignment, *i.e.,* estimating the homography between two planar projections of the same view but across different modalities such as map and satellite images (see Sec. 3.1.1 for formal definition), based on the LK method [46]. This method consists of two novel components:

- A new deep neural network was proposed to map images from different modalities into the same feature space where the LK method can align them.
- A new training algorithm was proposed as well by enforcing the local change on the loss landscape should be no less than a quadratic shape centered at the ground truth for any image pair, with no specific reason.

Surprisingly, when we evaluate DeepLK based on the public code[1], the proposed network cannot work well without the proposed training algorithm. This strongly motivate us to discover *the mysteries in the DeepLK training algorithm.*

Deep Reparametrization. Our first insight from DeepLK is that the deep neural network essentially maps the alignment problem into a much higher dimensional space by introducing a large amount of parameters. The high dimensional space provides the feasibility to reshape the loss landscape of the LK method. Such deep reparametrization has been used as a means of reformulating some problems such as shape analysis [11], super-resolution and denoising [8], while preserving the properties and constraints in the original problems. This insight at test time can be interpreted as

$$\min_{\omega\in\Omega} \ell(\omega; x) \xrightarrow[\text{via deep learning}]{\text{reparametrization}} \min_{\omega\in\Omega} \ell_f(\omega; x, \theta^*), \qquad (1)$$

where $x \in \mathcal{X}$ denotes the input data, ℓ denotes a nonconvex differentiable function (*e.g.,* the LK loss) parametrized by $\omega \in \Omega$, $f : \mathcal{X} \times \Theta \to \mathcal{X}$ denotes an auxiliary function presented by a neural network with learned weights $\theta^* \in \Theta$ (*e.g.,* the proposed network in DeepLK), and ℓ_f denotes the

[1] `https://github.com/placeforyiming/CVPR21-Deep-Lucas-Kanade-Homography`

loss with deep reparametrization (*e.g.,* the DeepLK loss). In this way, the learning problem is how to train the network so that the optimal solutions can be located using gradient descent (GD) given data.

Convex-like Loss Landscapes. Our second insight from DeepLK is that the learned loss landscape from their training algorithm tends to be convex-like (see their experimental results). This is an interesting observation, as it is evidenced in [39] that empirical more convex-like loss landscapes often return better performance. However, we cannot find any explicit explanation through the paper about the reason, which raises the following questions that we aim to address:

- Does the convex-like shape hold for any image pair?
- If so, why? Is there any guarantee on solutions?

Our Approach: Deep Star-Convexified Lucas-Kanade (PRISE). To mitigate the issue of poor local optimality in homography estimation, in this paper we propose a novel approach, namely PRISE, to enforce deep neural networks to approximately learn star-convex loss landscapes for the downstream tasks. Recently star-convexity [49] in nonconvex optimization has been attracting more and more attention [27, 30, 35, 38] because of its capability of finding near-optimal solutions based on GD with theoretical guarantees. Star-convex functions refer to a particular class of (typically) non-convex functions whose global optimum is visible from every point in a downhill direction. From this view, convexity is a special case of star-convexity. In the literature, however, most of the works focus on optimizing and analyzing star-convex functions, while learning such functions is hardly explored. In contrast, our PRISE imposes additional hinge losses, derived from the definition of star-convexity, on the learning objective during training. At test time, the nice convergence properties of star-convexity help find provably near-optimal solutions for the tasks using the LK method. We further show that DeepLK is a simplified and approximate algorithm of PRISE, and thus shares some properties with ours, but with worse performance.

Recently [78] have shown that stochastic gradient descent (SGD) will converge to global minimum in deep learning if the assumption of star-convexity in the loss landscapes hold. They validated this assumption (in a major part of training processes) empirically using relatively shallow networks and small-scale datasets by showing the classification training losses can converge to zeros. Nevertheless, we argue that this assumption may be too strong to hold in complex networks for challenging tasks, if without any additional imposition on learning. In our experiments we show that even we attempt to learn star-convex loss landscapes, the outputs at both training and test time are hardly perfect for complicated tasks.

Contributions. Our key contributions are listed as follows:

- We propose a novel PRISE method for multimodel image alignment by introducing (strongly) star-convex constraints into the network training, which is rarely explored in the literature of deep learning.
- We provide some analysis on the quality of the solutions from PRISE through star-convex loss landscapes.
- We demonstrate the state-of-the-art results on some benchmark datasets for multimodel image alignment with much better accuracy, especially when the pixel errors are small.

2. Related Work

Homography Estimation. Homography estimation is a classic task in computer vision. The feature-based methods [24, 74, 75] have existed for several decades but required similar contextual information to align the source and target images. To overcome this problem, researchers use deep neural networks [23, 36, 50, 76] to increase the alignment robustness between the source and template images. For instance, DHM [19] produces a distribution over quantized homographies to directly estimates the real-valued homography parameters. MHN [36] utilizes a multi-scale neural network to handle dynamic scenes. Since then, finding a combinatorial method from classical and deep learning approaches has become possible. Recent models such as CLKN [12], DeepLK [77] focus on learning a feature map for traditional Inverse Compositional Lucas-Kanade method on multimodal image pairs. Also, IHN [9] provides a correlation finding mechanism and iterative homography estimators across different scale to improve the performance of homography estimation without any untrainable part. A good survey can be found in [2].

Nonconvexity and Convexification. Nonconvexity is challenging in statistical learning where researchers proposed several regularized estimators [44, 45, 63] that can solve this issue partially. For deep learning or network training, such nonconvexity also brings serious trouble in optimization such as Adam [34]. Recently, the concept of convexification has started to be introduced into the training process [73]. Several works [47, 55, 64, 67] have demonstrated that the convex properties can be utilized in training a deep neural network whose loss landscape shows nonconvexity.

Adversarial Training. Adversarial training is one of the most effective strategies for improving robustness with adversarial data generation and model training. For the former, generative adversarial network (GAN) [25] and its variants [18] are classic deep neural networks that can be used to generate adversarial examples. For the latter, fast gradient sign method (FSGM) [26] and its variants [3, 20, 42] are widely used to train deep models. For instance, Shafahi *et al.* [56] proposed an algorithm that eliminates the overhead cost of generating adversarial examples by recycling the gradient information computed when updating model parameters. Wong *et al.* [69] demonstrated that it is possible to train empirically robust models using a much weaker and

cheaper adversary. Good surveys can be found in [6, 54].

Contrastive Learning. Recently, learning representations from unlabeled data in contrastive way [17, 28] has been one of the most competitive research field [5, 10, 13, 14, 16, 29, 31, 32, 40, 48, 52, 59, 61, 70]. Popular methods such as SimCLR [13] and Moco [29] apply the commonly used loss function InfoNCE [52] to learn latent representation that is beneficial to downstream tasks. Several theoretical studies show that contrastive loss optimizes data representations by aligning the same image's two views (positive pairs) while pushing different images (negative pairs) away on the hypersphere [4, 15, 66, 68]. In terms of applications there are a large amount of works in images [29, 40, 62, 79] and 3D point clouds [1, 21, 22, 33, 57, 60, 65, 71, 72], just to name a few. A good survey can be found in [37].

3. Deep Star-Convexified Lucas-Kanade

3.1. Preliminaries

3.1.1 Homography Estimation

Homography refers to a mapping between two planar projections of an image whose parameters are represented by a 3×3 transformation matrix in a homogenous coordinates space and need to be estimated. The LK method is one of the classic algorithms in computer vision for homography estimation between images. Its nonconvex objective can be formulated as follows:

$$\min_{\omega\in\Omega} \|x_t - g(x_s;\omega)\|_F^2, \tag{2}$$

where $x_s, x_t \in \mathcal{I}$ denote a source and target input images (equivalent to $x = \{x_s, x_t\}$ in Eq. 1), $\omega \in \Omega \subseteq \mathbb{R}^{3\times3}$ denotes the homography parameters, $g : \mathcal{I}\times\Omega \to \mathcal{I}$ denotes a nonconvex warping function, and $\|\cdot\|_F$ is the Frobenius norm. The LK algorithm uses GD to optimize Eq. 2.

3.1.2 DeepLK

Recently Zhao *et al.* [77] proposed a deep learning based LK method (DeepLK) that essentially rewrites Eq. 2 as follows:

$$\min_{\omega\in\Omega} \|f_t(x_t;\theta_t^*) - g(f_s(x_s;\theta_s^*);\omega)\|_F^2, \tag{3}$$

where functions $f_s : \mathcal{I}\times\Theta_s \to \mathcal{I}, f_t : \mathcal{I}\times\Theta_t \to \mathcal{I}$ denote two deep neural networks parametrized by the learned $\theta_s^* \in \Theta_s, \theta_t^* \in \Theta_t$, respectively (equivalent to $f = \{f_s, f_t\}, \theta^* = \{\theta_s^*, \theta_t^*\}, \Theta = \Theta_s \bigcup \Theta_t$ in Eq. 1), which transfer the source and target images into another two images. Then the original LK method can be directly applied to such transferred images for homography estimation with no change.

3.1.3 Star-Convexity

Definition 1 (Star-Convexity [38])**.** *A function $f : \mathbb{R}^n \to \mathbb{R}$ is* star-convex *if there is a global minimum $\omega^* \in \mathbb{R}^n$ such that for all $\lambda \in [0, 1]$ and $\omega \in \mathbb{R}^n$, it holds that*

$$f((1-\lambda)\omega^* + \lambda\omega) \le (1-\lambda)f(\omega^*) + \lambda f(\omega). \tag{4}$$

Definition 2 (Strong Star-Convexity [30])**.** *A differentiable function $f : \mathbb{R}^n \to \mathbb{R}$ is μ-*strongly star-convex *with constant $\mu > 0$ if there is a global minimum $\omega^* \in \mathbb{R}^n$ such that for $\forall\omega \in \mathbb{R}^n$, it holds*

$$f(\omega^*) \ge f(\omega) + \nabla f(\omega)^T(\omega^* - \omega) + \frac{\mu}{2}\|\omega^* - \omega\|_2^2, \tag{5}$$

where ∇ denotes the (sub)gradient operator, $(\cdot)^T$ denotes the matrix transpose operator, and $\|\cdot\|_2$ denotes the ℓ_2 norm. Note when $\mu = 0$ Eq. 5, will become equivalent to Eq. 4.

Lemma 1. *The following conditions hold iff a function $f : \mathbb{R}^n \to \mathbb{R}$ is μ-strongly star-convex, given a global minimum $\omega^* \in \mathbb{R}^n$ and $\forall\lambda \in [0, 1], \forall\omega \in \mathbb{R}^n$:*

$$f(\omega^*) \le f(\tilde{\omega}) - \frac{\mu}{2}\|\omega^* - \tilde{\omega}\|_2^2, \tag{6}$$

$$f(\tilde{\omega}) \le (1-\lambda)f(\omega^*) + \lambda f(\omega) - \frac{\lambda(1-\lambda)\mu}{2}\|\omega^* - \omega\|_2^2, \tag{7}$$

where $\tilde{\omega} = (1-\lambda)\omega^* + \lambda\omega$.

Proof. As illustrated in Fig. 1, a cut through ω^*, ω forms a convex shape if f is star-convex. Therefore, since $\nabla f(\omega^*) = \mathbf{0}$, Eq. 5 will lead to Eq. 6 by replacing ω with $\tilde{\omega}$ when switching the notations of ω^*, ω in the equation.

Letting $g(\omega) = f(\omega) - \frac{\mu}{2}\|\omega\|_2^2$, based on Eq. 5 we have $g(\omega^*) \ge g(\omega) + \nabla g(\omega)^T(\omega^* - \omega)$, *i.e.,* g is star-convex. Then based on g and Eq. 4, we can achieve Eq. 7. □

Eq. 6 implies that ω^* will be a (local) minimum if it holds for $\forall\omega, \lambda$. In fact, Lemma 1 discusses the (tight) strong star-convexity *with no gradients*. In our approach we will use this lemma to incorporate the strong star-convexity as constraints in network training.

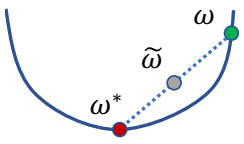


Figure 1. Geometric relations between $\omega^*, \omega, \tilde{\omega}$.

3.2. Approach

Geometric Constraints. Learning exact star-convex loss landscapes with the strong condition is challenging, even in very high dimensional spaces and in local regions. One common solution is to introduce slack variables as penalty to measure soft-margins. However, this may significantly break the smoothness of star-convex loss, and thus destroy the nice convergence property of gradient descent (see the results in

Fig. 7 in our experiments). Therefore, in order to preserve the smoothness, we consider two geometric constraints that have capability to improve the loss landscape smoothness at different levels, just in case that one is too strong to be learned properly. Ordered by the strength of each geometric constraint from weak to strong, they are:

- *A strong star-convexity constraint in Eq. 6:* This constraint implies that there exists a quadratic shape as the lower envelope of the loss landscape with minimum at ω^*. Meanwhile, it also guarantees that the loss at the ground-truth ω^* on the (local surface of) loss landscape will reach the (local) minimum, as requested by star-convexity.
- *A second strong star-convexity constraint in Eq. 7:* This constraint imposes strong convexity on all the curves that connect ω^* with any other point on the loss landscape.

The fundamental difference between the two strong star-convexity constraints is the positioning of ∇f, where Eq. 6 is posited at ω^* while Eq. 7 is posited at ω. As illustrated in Fig. 2, both constraints can lead to their own solution spaces for the same objective, but with an overlap where solutions will better approximate star-convexity. This insight motivates our formulation.

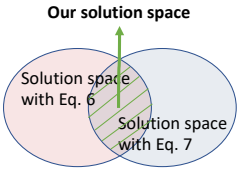


Figure 2. Illustration of solution spaces induced by the constraints.

Our Formulation. Let (x_i, ω_i^*) denote a training sample with data x_i (*e.g.,* image pairs for alignment) and its ground truth ω_i^*. To simplify our notations, in the sequel we will denote $h_\theta(\cdot) = \ell_f(\cdot; x_i, \theta), \tilde{\omega}_i = (1-\lambda)\omega_i^* + \lambda\omega_i$, and we wish to learn h_θ to be strongly star-convex. Now based on our considerations on geometric conditions, we are ready to formulate our learning framework as follows:

$$\min_\theta \sum_i \left\{ h_\theta(\omega_i^*) + \rho \mathbb{E}_{\omega_i \sim \mathcal{N}_{\omega_i^*}} \left\{ \max_{\lambda\in[0,1]} \epsilon_{\omega_i} + \max_{\lambda\in[0,1]} \xi_{\omega_i} \right\} \right\} \tag{8}$$

$$\text{s.t. } h_\theta(\omega_i^*) \le h_\theta(\tilde{\omega}_i) - \frac{\mu}{2}\|\omega_i^* - \tilde{\omega}_i\|_2^2 + \epsilon_{\omega_i}, \tag{9}$$

$$h_\theta(\tilde{\omega}_i) \le (1-\lambda)h_\theta(\omega_i^*) + \lambda h_\theta(\omega_i) - \frac{\lambda(1-\lambda)\mu}{2}\|\omega_i^* - \omega_i\|_2^2 + \xi_{\omega_i}, \tag{10}$$

$$\forall \epsilon_{\omega_i} \ge 0, \forall \xi_{\omega_i} \ge 0, \forall i,$$

where $\epsilon_{\omega_i}, \xi_{\omega_i}$ are the slack variables for hinge losses, $\mathcal{N}_{\omega_i^*}$ is a (local) neighborhood around ω_i^* where ω_i is sampled from, $\rho \ge 0, \mu \ge 0$ are predefined trade-off and surface sharpness parameters, respectively, and $\mathbb{E}$ denotes the expectation.

Contrastive Adversarial Training. Our approach is highly related to adversarial training and contrastive learning, since during training we try to create new fake samples $\tilde{\omega}_i, \omega_i$, compare their losses with $h_\theta(\omega^*)$, and solve a minimax problem defined in Eq. 8. The contrastive learning comes from the nature of strong star-convexity, leading to extra hinge losses. The adversarial training starts from finding the values of λ that return the maximum $\epsilon_{\omega_i}, \xi_{\omega_i}$, respectively. Note that here the values are allowed to be different for the two hinge losses. Both together aim to control the loss landscapes towards being star-convex.

Algorithm 1 PRISE: Deep Star-Convexified Lucas-Kanade

Input : training data $\{(x_i, \omega_i^*)\}$, LK loss function h_θ and a network architecture f, hyperparameters λ, μ, ρ

Output : network parameters θ^*

Randomly initialize θ;

repeat

  Randomly select a training image pair with its ground truth (x_i, ω_i^*);

  Randomly sample (multiple) $\omega_i \sim \mathcal{N}_{\omega_i^*}$;

  Update θ by solving Eq. 8 with strong star-convex constraints in Eqs. 9 and 10;

until *Converge or maximum number of iterations is reached*;

return $\theta^* \leftarrow \theta$;

Implementation. Eqs. 9 and 10 define a large pool of inequalities for each data point with varying ω_i and λ, where any inequality returns a hinge loss. To leverage our computational cost, motivated by the training algorithm for DeepLK we propose a similar *sampling* based training algorithm, as listed in Alg. 1[2]. Specifically,

- *Sampling from $\mathcal{N}_{\omega_i^*}$ for ω_i:* Same as stochastic gradient descent (SGD), we sample a fixed number of ω_i for each ground truth, and then compute the average.
- *Sampling from $[0,1]$ for λ:* We could solve the maximum problems using FSGM with careful parameter tuning. However, this will introduce huge computational burden, as the complexity will be proportional to the number of samples for ω_i times the number of data points (x_i, ω_i^*). Therefore, to address the computational complexity issue, we instead simply take λ as a predefined hyperparameter that are shared by $\epsilon_{\omega_i}, \xi_{\omega_i}$. We have evaluated the way of sampling multiple copies of λ and then choosing the maximum hinge losses with no sharing for learning. We observe that the results are very similar to those with the predefined one, but the training time is much longer.

As a demonstration, we take the network in DeepLK as our backbone and use the same LK loss as DeepLK for training. At test time, we substitute the learned network weights θ^* into the right side of Eq. 1 and solve the original non-convex image alignment problem using the LK method.

Star-Convexity *vs.* Convexity. Star-convexity enforces to learn one-point convexity [41], where we only impose the convexity at the ground truth within a local region. In contrast, convexity requires much more data points, making

[2] In our code we use batch-based implementation for fast computation.

the training of deep models much less efficient.

3.3. Analysis

Relations to DeepLK. In fact, DeepLK imposes the following two conditions on the minimization of the LK loss:

$$\begin{cases} h_\theta(\omega_i^*) \leq h_\theta(\omega_i) - \|\omega_i^* - \omega_i\|_2^2, \\ h_\theta(\tilde{\omega}_i) \leq h_\theta(\omega_i) - (1-\lambda^2)\|\omega_i^* - \omega_i\|_2^2. \end{cases} \tag{11}$$

Note that when $\lambda = 0$, these two inequalities will become the same. Below we will only discuss the lower equation in Eq. 11. Then we have the following lemma:

Lemma 2. *It holds for the RHS of Eqs. 11 and 10 that*

$$\begin{aligned}(1-\lambda)h_\theta(\omega_i^*) + \lambda h_\theta(\omega_i) - \frac{\lambda(1-\lambda)\mu}{2}\|\omega_i^* - \omega_i\|_2^2 \\ \overset{\mu\geq 2}{\leq} h_\theta(\omega_i) - (1-\lambda^2)\|\omega_i^* - \omega_i\|_2^2, \end{aligned} \tag{12}$$

with the same weights θ and the equality holds when $\mu = 2$.

Proof. With the help of Eq. 6 and simple algebra, we can easily prove this lemma. □

From this lemma, we can see that DeepLK potentially explores a larger solution space than our strong star-convexity. In other words, every solution returned from our PRISE would fall in the solution space of DeepLK. Therefore, we hypothesize that DeepLK may be able to learn some loss landscapes close to star-convex

Near-Optimal Solutions. [30] has shown that the GD based algorithms can find near-optimal solutions for the optimization of star-convex functions. To better see this intuitively, we can have the following inequality based on Eq. 6:

$$\|\bar{\omega}^* - \bar{\omega}\|^2 \leq \frac{2}{\mu}\Big[h_{\theta^*}(\bar{\omega}) - h_{\theta^*}(\bar{\omega}^*)\Big], \tag{13}$$

where $\bar{\omega}^*$ denotes the ground truth for a test data point $\bar{x}$, $\bar{\omega}$ denotes the prediction based on a network with learned weights θ^*, and $\mu \geq 0$ is a constant. For image alignment with the LK loss, $h_{\theta^*}(\bar{\omega}^*) = 0$ and thus a smaller $h_{\theta^*}(\bar{\omega})$ implies a better solution, which is minimized using the LK loss in training. In fact, however, the distance is upper-bounded by the *contrastive loss*, which contributes to the hinge losses that are appended to the LK loss in our formulation. This observation indicates that the contribution of the hinge losses to a well-trained network may be higher than the LK loss (see Fig. 3 for more details).

4. Experiments

Datasets. We exactly follow the experimental settings in DeepLK [77]. We select an image from each dataset and resize it to 196×196 pixels as input. Then we randomly perturb 4 points in the four corner boxes with size of 64×64 and resize the chosen region to a 128×128 template image. We implement the same data generation strategy on three different datasets as follows:

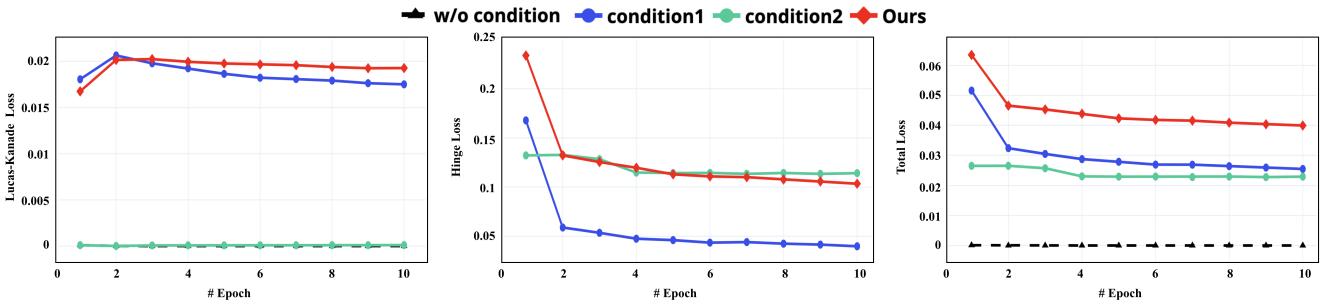
- *MSCOCO [43]:* This is a benchmark dataset in computer vision, including homography estimation, with various foreground and background information. 6K images are sampled from the validation set as our test set.
- *GoogleEarth [77]:* This is a high-resolution cross-season satellite dataset consisting of about 8K training images and 850 test images. Homography estimation on this dataset is challenging as the textural differences between different images are small (compared with the natural images in MSCOCO), and they can easily confuse the LK algorithm which tries to catch the change in grey scale. It has been demonstrated in DeepLK that many recent works on homography estimation failed on this dataset.
- *Google Maps and Satellite (GoogleMap for short) [77]:* This dataset provides multimodal inputs for query and template images, consisting of about 8K training image pairs from Google Map and Google Satellite at the same locations and 888 test pairs. We use the satellite images as the queries and the google map images as the templates to find the homography change from the satellite data to map data. Many models such as DHN [19] and MHN [36] failed to work on this dataset.

Baseline Algorithms. We train DHM [19] and MHN [36] from scratch with the best hyperparameters. We use the pretrained model for DeepLK [77] directly. In addition, we use the pretrained models for CLKN [12] and fine-tune it on MSCOCO, GoogleEarth, and GoogleMap to fix the domain gap. Also we compare our approach with a classical algorithm SIFT+RANSAC.

Training & Testing Protocols. Following the consistent setting for each dataset in both training and testing, we use the same resolution for the source and target images as the standard datasets. For training, we train our model with best hyperparameters on each dataset for 10 times with random initialization of network weights. For testing, we conduct evaluation on the PEs under different thresholds. We report our results in terms of mean and standard deviation over 10 trials. All the experiments are done using an Nvidia RTX6000 GPU server

Our Implementation Details. To demonstrate the improvement of our approach, we modify the implementation of DeepLK. Specifically,

- *Network architecture:* We employ the same network architecture in DeepLK. The network has three identical stages with the same sub-networks. In each stage, it is a siamese design sharing weight with the source and target images. There are 3 residual blocks in each siamese architecture with 64 convolutional filters. The network downsamples



(a) LK loss (b) Hinge loss (c) Total loss

Figure 3. Comparison on the training loss at Stage 3 on GoogleEarth.

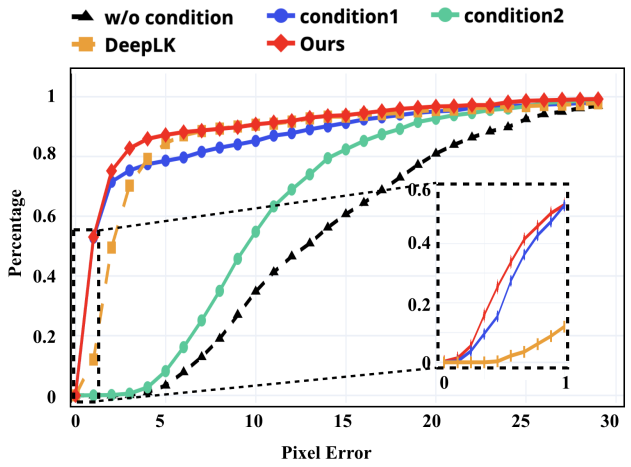


Figure 4. Performance comparison on GoogleEarth using the default hyperparameter setting.

the output feature maps using the stride of 2.

- *Training:* We train the network in a stacked way (*i.e.,* Stage 1 first, then Stage 2 and finally Stage 3), not end-to-end. That is, once the previous stage is fully trained, the current stage will start training using the output feature maps from the previous stage as input. We train each stage in the network for 10 epochs with a batch size of 4, a constant learning rate 10^{-5}, and weight decay 0.05. Except μ, λ, ρ that are determined by grid search, all the rest hyperparameters keep the same as used in DeepLK. We employ Adam [34] as our optimizer. By default, we use $\mu = 2, \lambda = 0.9, \rho = 0.1, \#sample = 2$ for MSCOCO, $\mu = 4, \lambda = 0.5, \rho = 0.2, \#sample = 4$ for GoogleEarth, and $\mu = 2, \lambda = 0.5, \rho = 0.2, \#sample = 4$ for GoogleMap, where $\#sample$ denotes the number of samples for ω_i (see Alg. 1).
- *Inference:* Our inference follows the coarse-to-fine strategy that fits the iterative updated methods in the classic LK algorithm. That is, both source and target images go through the learned network to extract three feature maps, one per stage, and then the classic LK algorithm is applied to each feature map, starting from Stage 3 up to Stage 1 in a backward manner. The homography parameters will be upscale with a factor of 2 sequentially since the resolutions of feature maps are different, until Stage 1 is done. MHN [36] is used for initializing homography parameters.

Evaluation Metric. We use the same evaluation metric as in recent works, Success Rate (SR) *vs.* Pixel Error (PE), to compare the performance of each algorithm. PE measures the average L_2 distance between the 4 ground-truth perturbation points and the 4 output point location predictions (without quantization) from an algorithm, correspondingly. Then the percentages of the testing image pairs whose PEs are smaller than certain thresholds, *i.e.,* SR, are computed to compare the performance of different approaches.

4.1. Ablation Study

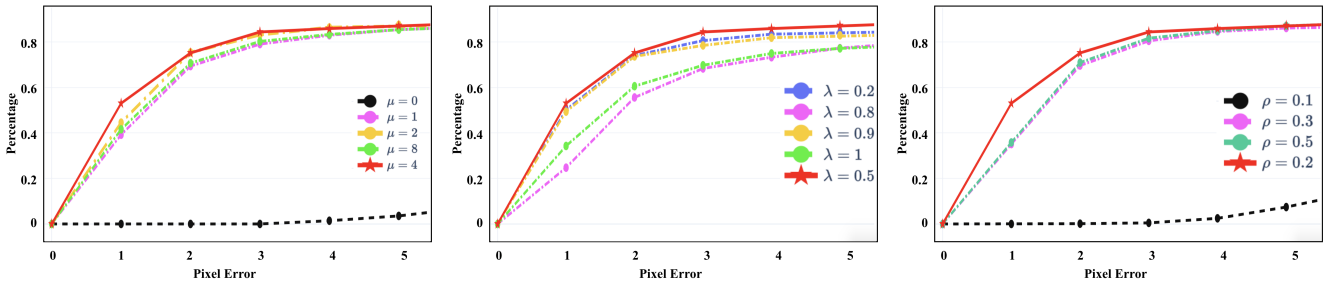
4.1.1 Effects of Strong Star-Convexity Constraints

Fig. 3 illustrates our comparisons on the training loss at Stage 3, where (a) shows the loss for the LK algorithm, (b) shows the combination loss from the extra hinge losses, and (c) shows the weighted sum of the two losses in (a) and (b) as our objective in Eq. 8. Here we refer to Eq. 9 as Condition 1, and Eq. 10 as Condition 2. From this figure, we can see that

- The LK loss and hinge loss are balanced well at the same scale, no dominance scenarios occurring. All the losses tend to converge over the epochs.
- From Fig. 3 (c), without any hinge loss, training with the original LK loss alone is very easy to overfit.
- Condition 1 alone seems much more important than Condition 2, because from Fig. 3 (a) it can prevent the loss from overfitting (while Condition 2 cannot) and from Fig. 3 (b) it can lead to a much lower loss.

Such observations partially support our analysis in Sec. 3.3, where narrower solution spaces seem not to help regularize the learning (*i.e.,* Lemma 2), and the contrastive loss contributes more to the learning (*i.e.,* Eq. 13) as it leads to reduce the total loss, even though the LK loss increases.

We illustrate our performance comparison in Fig. 7. Overall, the performance is consistent with the LK training losses



(a) (b) (c)

Figure 5. Pixel error *vs.* various hyperparameters in Eqs. 8-10 on GoogleEarth.

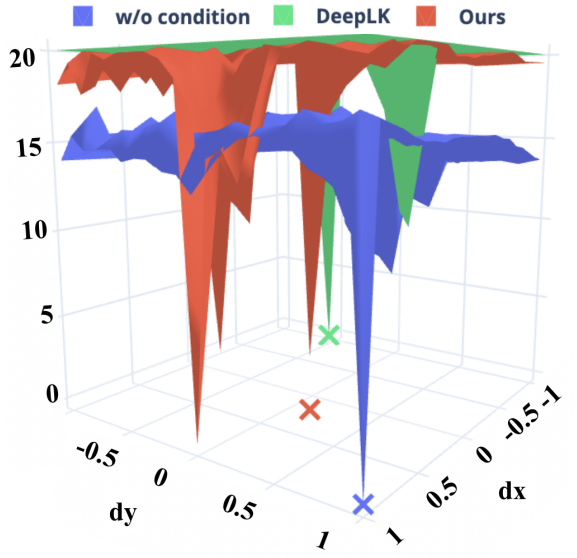


Figure 6. Visualization of the loss landscapes of the plain model, DeepLK and PRISE by using a test pair of source image and target image randomly selected from GoogleEarth, and the $\times$'s on the dx-dy plane denote the locations of the homography estimation results from the three methods around the ground truth at $(0, 0)$.

that do not overfit. As we see, the method with Condition 2 only shows similar performance to the one without using any condition, and they are hard to converge in all PE evaluations. The method with Condition 1 only show significant improvements over the one with Condition 2 in all the evaluations, and outperforms DeepLK in smaller PE evaluations but becomes worse in larger PE evaluations. Using both conditions our PRISE further boosts the performance, and achieves the best among all the competitors, thanks to the learning within the overlap of solution spaces defined by both strongly star-convex constraints.

4.1.2 Effects of Hyperparameters

Fig. 5 illustrates the comparisons among different hyperparameter settings using PRISE. We only evaluate one hyperparameter once while fixing the others as the default values. Overall, our approach is very robust to different hyperparameters. Specifically, Fig. 5 (a) demonstrates the necessity of strong star-convexity, which verifies our insight on learning a unique minimum within a local region rather than several minima (*i.e.,* not strong). When μ is sufficiently large, the performance gaps are small. Fig. 5 (b) shows that the midpoint choice (*i.e.,* $\lambda = 0.5$) indeed provides good performance, which intuitively follows the classic results of midpoint convex and continuous functions being convex [53]. Fig. 5 (c) shows that when ρ is small, the hinge loss cannot stop the overfitting, and when it increases to a sufficiently large number, the performance will be improved significantly and stably. All the values for ρ make good balance between the LK loss and the hinge loss.

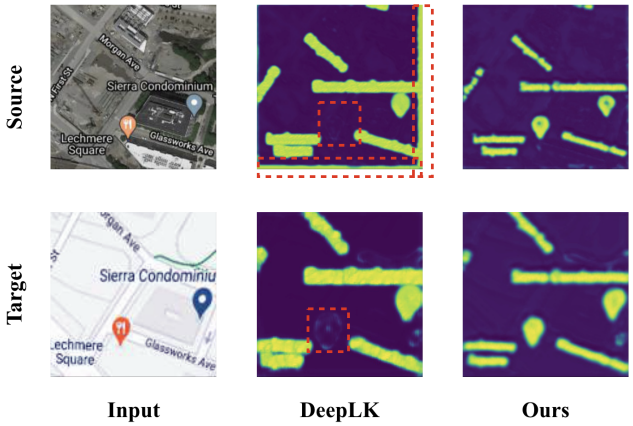


Figure 7. Feature map comparison on GoogleMap dataset, with boxes emphasizing the differences between DeepLK and ours.

4.1.3 Visualization

Fig. 6 visualizes our loss landscape comparison results based on the outputs from Stage 3, where two random entries in the ground-truth homography matrix of the selected image pair are manipulated while the other entries are fixed.

- PRISE can generate a locally star-convex shape around the ground truth, and accordingly return an estimation that is

Table 1. Performance comparison on MSCOCO, GoogleEarth, and GoogleMap.

Dataset	Method	PE<0.1	PE<0.5	PE<1	PE<3	PE<5	PE<10	PE<20
MSCOCO	SIFT+RANSAC	0.00	4.70	68.32	84.21	90.32	95.26	96.55
	SIFT+MAGSAC [7]	0.00	3.66	76.27	93.26	94.22	95.32	97.26
	LF-Net [51]	5.60	8.62	14.20	23.00	78.88	90.18	95.45
	LocalTrans [58]	38.24	**87.25**	96.45	98.00	98.72	99.25	**100.00**
	DHM [19]	0.00	0.00	0.87	3.48	15.27	98.22	99.96
	MHN [36]	0.00	4.58	81.99	95.67	96.02	98.45	98.70
	CLKN [12]	35.24	83.25	83.27	94.26	95.75	97.52	98.46
	DeepLK [77]	17.16	72.25	92.81	96.76	97.67	98.92	99.03
	PRISE	**52.77** $\pm$ 12.45	83.27 $\pm$ 5.21	**97.29** $\pm$ 1.82	**98.44** $\pm$ 1.06	**98.76** $\pm$ 0.08	**99.31** $\pm$ 0.53	99.33 $\pm$ 1.84
GoogleEarth	SIFT+RANSAC	0.18	3.42	8.97	23.09	41.32	50.36	59.88
	SIFT+MAGSAC [7]	0.00	0.00	1.88	2.70	3.25	10.03	45.29
	DHM [19]	0.00	0.02	1.46	2.65	5.57	25.54	90.32
	MHN [36]	0.00	3.42	4.56	5.02	8.99	59.90	93.77
	CLKN [12]	**0.27**	2.88	3.45	4.24	4.32	8.77	75.00
	DeepLK [77]	0.00	3.50	12.01	70.20	84.45	90.57	95.52
	PRISE	0.24 $\pm$ 1.83	**25.44** $\pm$ 1.21	**53.00** $\pm$ 1.54	**82.69** $\pm$ 1.07	**87.16** $\pm$ 1.09	**90.69** $\pm$ 0.73	**96.70** $\pm$ 0.54
GoogleMap	SIFT+RANSAC	0.00	0.00	0.00	0.00	0.00	2.74	3.44
	SIFT+MAGSAC [7]	0.00	0.00	0.00	0.00	0.00	0.15	2.58
	DHM [19]	0.00	0.00	0.00	1.20	3.43	6.99	78.89
	MHN [36]	0.00	0.34	0.45	0.50	3.50	35.69	93.77
	CLKN [12]	0.00	0.00	0.00	1.57	1.88	8.67	22.45
	DeepLK [77]	0.00	2.25	16.80	61.33	73.39	83.20	93.80
	PRISE	**17.47** $\pm$ 2.44	**48.13** $\pm$ 12.00	**56.93** $\pm$ 3.45	**76.21** $\pm$ 2.43	**80.04** $\pm$ 5.55	**86.13** $\pm$ 0.47	**94.02** $\pm$ 1.66

much closer to the ground truth than the other two, which follows what we expect.
- All the three methods cannot generate smooth shapes over relatively large areas in the parameter space.
- The LK loss alone can achieve lower values, but this does not help estimation, while DeepLK and PRISE can achieve similar values overall.

Fig. 7 illustrates the differences in feature maps generated by DeepLK and our PRISE, where PRISE learns the feature maps more accurately, thus leading to better performance.

4.2. State-of-the-art Comparison

We list our comparison results in Table 1. Clearly, our PRISE significantly improves DeepLK in all the cases. Overall, PRISE achieves the best among all the competitors with dramatically performance gaps, especially when PE is small. Recall that GoogleMap is specifically designed for multi-model image alignment, and the superior performance of PRISE better demonstrates its usage in the application. We also report the standard deviation (std) for PRISE to show that our improvements are statistically significant, and very often the std is marginal to the mean.

5. Conclusion

Motivated by a recent work DeepLK, in this paper we propose a novel approach for multimodel image alignment, namely, Deep Star-Convexified Lucas Kanade (PRISE), to find near-optimal solutions. Our idea is to reparametrize the loss landscapes of the LK method to be star-convex using deep learning. To this end, we introduce extra hinge losses based on the definition of strong star-convexity, and impose them on the original LK loss to enforce learning star-convex loss landscapes (approximately). This leads to a minimax problem that is solvable using adversarial training. Further, to leverage the computational cost, we propose an efficient sampling based algorithm to train PRISE. We also provide some analysis on the homography estimation results from PRISE. We finally demonstrate our approach on three benchmark datasets for image alignment and show the state-of-the-art results, especially when the PE is small.

We are aware of several very recent works that report the performance on these three benchmarks, *e.g.,* Iterative Homography Network (IHN) [9]. Unfortunately so far we cannot reproduce the results in the paper using the public code. We will try to add such new results when they are ready. Note that, however, our approach provides a general learning framework that can fit to not only the DeepLK network but also other existing networks such as IHN. In the future we will adapt our learning framework to train other networks with strongly star-convex constraints.

Acknowledgement

Y. Zhang, X. Huang, and Z. Zhang were all supported partially by NSF CCF-2006738.

References

[1] Mohamed Afham, Isuru Dissanayake, Dinithi Dissanayake, Amaya Dharmasiri, Kanchana Thilakarathna, and Ranga Rodrigo. Crosspoint: Self-supervised cross-modal contrastive learning for 3d point cloud understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9902–9912, 2022.

[2] Anubhav Agarwal, CV Jawahar, and PJ Narayanan. A survey of planar homography estimation techniques. *Centre for Visual Information Technology, Tech. Rep. IIIT/TR/2005/12*, 2005.

[3] Maksym Andriushchenko and Nicolas Flammarion. Understanding and improving fast adversarial training. *arXiv preprint arXiv:2007.02617*, 2020.

[4] Sanjeev Arora, Hrishikesh Khandeparkar, Mikhail Khodak, Orestis Plevrakis, and Nikunj Saunshi. A theoretical analysis of contrastive unsupervised representation learning. *arXiv preprint arXiv:1902.09229*, 2019.

[5] Philip Bachman, R Devon Hjelm, and William Buchwalter. Learning representations by maximizing mutual information across views. *Advances in neural information processing systems*, 32, 2019.

[6] Tao Bai, Jinqi Luo, Jun Zhao, Bihan Wen, and Qian Wang. Recent advances in adversarial training for adversarial robustness. *arXiv preprint arXiv:2102.01356*, 2021.

[7] Daniel Barath, Jiri Matas, and Jana Noskova. MAGSAC: marginalizing sample consensus. In *Conference on Computer Vision and Pattern Recognition*, 2019.

[8] Goutam Bhat, Martin Danelljan, Fisher Yu, Luc Van Gool, and Radu Timofte. Deep reparametrization of multi-frame super-resolution and denoising. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2460–2470, 2021.

[9] Si-Yuan Cao, Jianxin Hu, Zehua Sheng, and Hui-Liang Shen. Iterative deep homography estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1879–1888, June 2022.

[10] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. *Advances in Neural Information Processing Systems*, 33:9912–9924, 2020.

[11] Elena Celledoni, Helge Glöckner, Jørgen Riseth, and Alexander Schmeding. Deep learning of diffeomorphisms for optimal reparametrizations of shapes. *arXiv preprint arXiv:2207.11141*, 2022.

[12] Che-Han Chang, Chun-Nan Chou, and Edward Y Chang. Clkn: Cascaded lucas-kanade networks for image alignment. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2213–2221, 2017.

[13] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020.

[14] Ting Chen, Simon Kornblith, Kevin Swersky, Mohammad Norouzi, and Geoffrey E Hinton. Big self-supervised models are strong semi-supervised learners. *Advances in neural information processing systems*, 33:22243–22255, 2020.

[15] Ting Chen, Calvin Luo, and Lala Li. Intriguing properties of contrastive losses. *Advances in Neural Information Processing Systems*, 34, 2021.

[16] Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*, 2020.

[17] Sumit Chopra, Raia Hadsell, and Yann LeCun. Learning a similarity metric discriminatively, with application to face verification. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pages 539–546. IEEE, 2005.

[18] Antonia Creswell, Tom White, Vincent Dumoulin, Kai Arulkumaran, Biswa Sengupta, and Anil A Bharath. Generative adversarial networks: An overview. *IEEE signal processing magazine*, 35(1):53–65, 2018.

[19] Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Deep image homography estimation. *arXiv preprint arXiv:1606.03798*, 2016.

[20] Yinpeng Dong, Fangzhou Liao, Tianyu Pang, Hang Su, Jun Zhu, Xiaolin Hu, and Jianguo Li. Boosting adversarial attacks with momentum. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 9185–9193, 2018.

[21] Bi'an Du, Xiang Gao, Wei Hu, and Xin Li. Self-contrastive learning with hard negative sampling for self-supervised point cloud learning. In *Proceedings of the 29th ACM International Conference on Multimedia*, pages 3133–3142, 2021.

[22] Benjamin Eckart, Wentao Yuan, Chao Liu, and Jan Kautz. Self-supervised learning on 3d point clouds by learning discrete generative models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8248–8257, 2021.

[23] Farzan Erlik Nowruzi, Robert Laganiere, and Nathalie Japkowicz. Homography estimation from image pairs with hierarchical convolutional networks. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 913–920, 2017.

[24] Jun Fu, Jing Liu, Yuhang Wang, Yong Li, Yongjun Bao, Jinhui Tang, and Hanqing Lu. Adaptive context network for scene parsing. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6748–6757, 2019.

[25] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020.

[26] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014.

[27] Robert Gower, Othmane Sebbouh, and Nicolas Loizou. Sgd for structured nonconvex functions: Learning rates, minibatching and interpolation. In *International Conference on Artificial Intelligence and Statistics*, pages 1315–1323. PMLR, 2021.

[28] Raia Hadsell, Sumit Chopra, and Yann LeCun. Dimensionality reduction by learning an invariant mapping. In *2006*

IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06), volume 2, pages 1735–1742. IEEE, 2006.

[29] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9729–9738, 2020.

[30] Oliver Hinder, Aaron Sidford, and Nimit Sohoni. Near-optimal methods for minimizing star-convex functions and beyond. In *Proceedings of Thirty Third Conference on Learning Theory*, pages 1894–1938, 2020.

[31] R Devon Hjelm, Alex Fedorov, Samuel Lavoie-Marchildon, Karan Grewal, Phil Bachman, Adam Trischler, and Yoshua Bengio. Learning deep representations by mutual information estimation and maximization. *arXiv preprint arXiv:1808.06670*, 2018.

[32] Ashish Jaiswal, Ashwin Ramesh Babu, Mohammad Zaki Zadeh, Debapriya Banerjee, and Fillia Makedon. A survey on contrastive self-supervised learning. *Technologies*, 9(1):2, 2020.

[33] Li Jiang, Shaoshuai Shi, Zhuotao Tian, Xin Lai, Shu Liu, Chi-Wing Fu, and Jiaya Jia. Guided point contrastive learning for semi-supervised point cloud semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6423–6432, 2021.

[34] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[35] Ilya A Kuruzov and Fedor S Stonyakin. Sequential subspace optimization for quasar-convex optimization problems with inexact gradient. In *International Conference on Optimization and Applications*, pages 19–33. Springer, 2021.

[36] Hoang Le, Feng Liu, Shu Zhang, and Aseem Agarwala. Deep homography estimation for dynamic scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7652–7661, 2020.

[37] Phuc H Le-Khac, Graham Healy, and Alan F Smeaton. Contrastive representation learning: A framework and review. *IEEE Access*, 8:193907–193934, 2020.

[38] Jasper CH Lee and Paul Valiant. Optimizing star-convex functions. In *2016 IEEE 57th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 603–614. IEEE, 2016.

[39] Hao Li, Zheng Xu, Gavin Taylor, Christoph Studer, and Tom Goldstein. Visualizing the loss landscape of neural nets. *Advances in neural information processing systems*, 31, 2018.

[40] Junnan Li, Pan Zhou, Caiming Xiong, and Steven CH Hoi. Prototypical contrastive learning of unsupervised representations. *arXiv preprint arXiv:2005.04966*, 2020.

[41] Yuanzhi Li and Yang Yuan. Convergence analysis of two-layer neural networks with relu activation. *Advances in Neural Information Processing Systems*, 30:597–607, 2017.

[42] Jiadong Lin, Chuanbiao Song, Kun He, Liwei Wang, and John E Hopcroft. Nesterov accelerated gradient and scale invariance for adversarial attacks. *arXiv preprint arXiv:1908.06281*, 2019.

[43] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

[44] Po-Ling Loh and Martin J Wainwright. Regularized m-estimators with nonconvexity: Statistical and algorithmic theory for local optima. *arXiv preprint arXiv:1305.2436*, 2013.

[45] Po-Ling Loh and Martin J Wainwright. Regularized m-estimators with nonconvexity: Statistical and algorithmic theory for local optima. *The Journal of Machine Learning Research*, 16(1):559–616, 2015.

[46] Bruce D Lucas and Takeo Kanade. An iterative image registration technique with an application to stereo vision. In *Proceedings of Imaging Understanding Workshop*, pages 121–130, 1981.

[47] Yuanqi Mao, Michael Szmuk, and Behçet Açıkmeşe. Successive convexification of non-convex optimal control problems and its convergence properties. In *2016 IEEE 55th Conference on Decision and Control (CDC)*, pages 3636–3641. IEEE, 2016.

[48] Ishan Misra and Laurens van der Maaten. Self-supervised learning of pretext-invariant representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6707–6717, 2020.

[49] Yurii Nesterov and Boris T Polyak. Cubic regularization of newton method and its global performance. *Mathematical Programming*, 108(1):177–205, 2006.

[50] Ty Nguyen, Steven W Chen, Shreyas S Shivakumar, Camillo Jose Taylor, and Vijay Kumar. Unsupervised deep homography: A fast and robust homography estimation model. *IEEE Robotics and Automation Letters*, 3(3):2346–2353, 2018.

[51] Yuki Ono, Eduard Trulls, Pascal Fua, and Kwang Moo Yi. Lf-net: Learning local features from images. *Advances in neural information processing systems*, 31, 2018.

[52] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

[53] HX Phu. Six kinds of roughly convex functions. *Journal of optimization theory and applications*, 92(2):357–375, 1997.

[54] Zhuang Qian, Kaizhu Huang, Qiu-Feng Wang, and Xu-Yao Zhang. A survey of robust adversarial training in pattern recognition: Fundamental, theory, and methodologies. *arXiv preprint arXiv:2203.14046*, 2022.

[55] S Reddi, Manzil Zaheer, Devendra Sachan, Satyen Kale, and Sanjiv Kumar. Adaptive methods for nonconvex optimization. In *Proceeding of 32nd Conference on Neural Information Processing Systems (NIPS 2018)*, 2018.

[56] Ali Shafahi, Mahyar Najibi, Amin Ghiasi, Zheng Xu, John Dickerson, Christoph Studer, Larry S Davis, Gavin Taylor, and Tom Goldstein. Adversarial training for free! *arXiv preprint arXiv:1904.12843*, 2019.

[57] Huixiang Shao, Zhijiang Zhang, Xiaoyu Feng, and Dan Zeng. Scrnet: A spatial consistency guided network using contrastive learning for point cloud registration. *Symmetry*, 14(1):140, 2022.

[58] Ruizhi Shao, Gaochang Wu, Yuemei Zhou, Ying Fu, Lu Fang, and Yebin Liu. Localtrans: A multiscale local transformer network for cross-resolution homography estimation. In *IEEE Conference on Computer Vision (ICCV 2021)*, 2021.

[59] Kihyuk Sohn. Improved deep metric learning with multi-class n-pair loss objective. *Advances in neural information processing systems*, 29, 2016.

[60] Liyao Tang, Yibing Zhan, Zhe Chen, Baosheng Yu, and Dacheng Tao. Contrastive boundary learning for point cloud segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8489–8499, 2022.

[61] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. In *European conference on computer vision*, pages 776–794. Springer, 2020.

[62] Yonglong Tian, Chen Sun, Ben Poole, Dilip Krishnan, Cordelia Schmid, and Phillip Isola. What makes for good views for contrastive learning? *Advances in Neural Information Processing Systems*, 33:6827–6839, 2020.

[63] Hoang Duong Tuan and Pierre Apkarian. Low nonconvexity-rank bilinear matrix inequalities: algorithms and applications in robust controller and structure designs. *IEEE Transactions on Automatic Control*, 45(11):2111–2117, 2000.

[64] Sujit Vettam and Majnu John. Regularized deep learning with nonconvex penalties. *arXiv preprint arXiv:1909.05142*, 2019.

[65] Di Wang, Lulu Tang, Xu Wang, Luqing Luo, and Zhi-Xin Yang. Improving deep learning on point cloud by maximizing mutual information across layers. *Pattern Recognition*, 131:108892, 2022.

[66] Feng Wang and Huaping Liu. Understanding the behaviour of contrastive loss. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2495–2504, 2021.

[67] Huachuan Wang and James Ting-Ho Lo. Adaptively solving the local-minimum problem for deep neural networks. *arXiv preprint arXiv:2012.13632*, 2020.

[68] Tongzhou Wang and Phillip Isola. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *International Conference on Machine Learning*, pages 9929–9939. PMLR, 2020.

[69] Eric Wong, Leslie Rice, and J Zico Kolter. Fast is better than free: Revisiting adversarial training. *arXiv preprint arXiv:2001.03994*, 2020.

[70] Zhirong Wu, Yuanjun Xiong, Stella X Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3733–3742, 2018.

[71] Siming Yan, Zhenpei Yang, Haoxiang Li, Li Guan, Hao Kang, Gang Hua, and Qixing Huang. Implicit autoencoder for point cloud self-supervised representation learning. *arXiv preprint arXiv:2201.00785*, 2022.

[72] Cheng-Kun Yang, Yung-Yu Chuang, and Yen-Yu Lin. Unsupervised point cloud object co-segmentation by co-contrastive learning and mutual attention sampling. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7335–7344, 2021.

[73] Heng Yang, Pasquale Antonante, Vasileios Tzoumas, and Luca Carlone. Graduated non-convexity for robust spatial perception: From non-minimal solvers to global outlier rejection. *IEEE Robotics and Automation Letters*, 5(2):1127–1134, 2020.

[74] Yuanxin Ye, Lorenzo Bruzzone, Jie Shan, Francesca Bovolo, and Qing Zhu. Fast and robust matching for multimodal remote sensing image registration. *IEEE Transactions on Geoscience and Remote Sensing*, 57(11):9059–9070, 2019.

[75] Yuanxin Ye, Jie Shan, Lorenzo Bruzzone, and Li Shen. Robust registration of multimodal remote sensing images based on structural similarity. *IEEE Transactions on Geoscience and Remote Sensing*, 55(5):2941–2958, 2017.

[76] Jirong Zhang, Chuan Wang, Shuaicheng Liu, Lanpeng Jia, Nianjin Ye, Jue Wang, Ji Zhou, and Jian Sun. Content-aware unsupervised deep homography estimation. In *European Conference on Computer Vision*, pages 653–669. Springer, 2020.

[77] Yiming Zhao, Xinming Huang, and Ziming Zhang. Deep lucas-kanade homography for multimodal image alignment. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15950–15959, 2021.

[78] Yi Zhou, Junjie Yang, Huishuai Zhang, Yingbin Liang, and Vahid Tarokh. SGD converges to global minimum in deep learning via star-convex path. In *International Conference on Learning Representations*, 2019.

[79] Roland S Zimmermann, Yash Sharma, Steffen Schneider, Matthias Bethge, and Wieland Brendel. Contrastive learning inverts the data generating process. In *International Conference on Machine Learning*, pages 12979–12990. PMLR, 2021.