SDGMNet: Statistic-based Dynamic Gradient Modulation for Local Descriptor Learning

Jiayi Ma, *Senior Member, IEEE*, and Yuxin Deng

***Abstract*—Modifications on triplet loss that rescale the back-propagated gradients of special pairs have made significant progress on local descriptor learning. However, current gradient modulation strategies are mainly static so that they would suffer from changes of training phases or datasets. In this paper, we propose a dynamic gradient modulation, named SDGMNet, to improve triplet loss for local descriptor learning. The core of our method is formulating modulation functions with statistical characteristics which are estimated dynamically. Firstly, we perform deep analysis on back propagation of general triplet-based loss and introduce included angle for distance measure. On this basis, auto-focus modulation is employed to moderate the impact of statistically uncommon individual pairs in stochastic gradient descent optimization; probabilistic margin cuts off the gradients of proportional Siamese pairs that are believed to reach the optimum; power adjustment balances the total weights of negative pairs and positive pairs. Extensive experiments demonstrate that our novel descriptor surpasses previous state-of-the-arts on standard benchmarks including patch verification, matching and retrieval tasks. Our code is available at https://github.com/jiayi-ma/SDGMNet.**

***Index Terms*—Local descriptor learning, dynamic gradient modulation, statistical characteristics.**

I. Introduction

EVALUATING local correspondences of images is a fundamental problem in many computer vision tasks, such as visual localization [36], [61], image registration [26] and retrieval [27]. For this purpose, the classic two-stage pipeline including keypoint detection and patch description was proposed decades ago. Although promising end-to-end methods [58], [7], [8], [34], [45], [24] sprung up in recent years, the traditional pipeline remains competitive due to its robustness and efficiency in practice [25], [16]. Moreover, deep local descriptors [42], [2], [18], [47], [60], [13], [17], [23], [54], [59], [48], [51], [9], [46], [52], [56] that are learned with deep neural networks have noticeably outperformed their hand-crafted counterparts, *e.g.*, Scale-Invariant Feature Transform (SIFT) [22], [19], and boosted the performance of the two-stage pipeline. Therefore, local descriptors based on deep learning merit deeper study.

Benefiting from the great potentials of Deep Neural Network (DNN), deep local descriptors dispense with heuristic designs to acquire invariance as early efforts [22], [35] did. Overall, local descriptor learning is exactly a branch of metric learning [30]. Specifically, this task aims to encode local patches into discriminative descriptors, and then predict whether pairs of patches are matching or not according to distances between descriptors. To train the encoder, we need to minimize the distance of matching/positive pairs and maximize non-matching/negative ones in the loss function. To this end, various loss functions taking pairs as basic units are proposed, such as pair-wise loss [18], triplet loss [2], [28], n-pair loss [47] and ranking loss [13]. Particularly, HardNet [28] constructs a potent triplet loss by mining hard negatives in L2Net [47] batch. Recent works mainly focus on modifying this loss due to its simplicity and superiority.

Besides imposing extra regularization [60], [48] or resampling patches [9], most modifications [17], [54], [59], [51], [46] devote to modulating gradients of pairs according to their hardness of discriminating or identifying. Specifically, if an individual positive pair is too distant to identify, its back-propagated gradients should be weighted more largely during optimization. In contrast, the weight for a hard negative pair that is closer should be larger. Moreover, the hardness of Siamese pairs that share the same anchor also deserves attentions. This kind of hardness can be similarly defined with relative distance. These principles for gradient modulation are so-called hard example mining (HEM). We briefly illustrate them in Fig. 1. HEM also dominates the design of pair-based loss functions in other metric learning tasks [31], [50], [20], [49], [6], [53], [44], [14]. Their successes demonstrate the significance of gradient modulation. However, most modulations are static. The values of the modulation functions depend on individual pairs or Siamese pairs, but do not involve the training phase or the global information. Such modulations suffer from changes of training phases and datasets. Thus, standing on modulating the gradients of individual pairs and Siamese pairs, we concentrate on proposing a dynamic modulation for local descriptor learning. Drawing on global statistical characteristics that are dynamically estimated is promising. Statistics can provide global information that varies over time and datasets, which makes the learning adaptive.

While we are formulating a statistic-based dynamic gradient modulation, more details should be explored. Firstly, related works [46], [62] indicate that once a specific metric, *e.g.*, L_2 distance or inner product is chosen, an implicit modulation harbored in deep back propagation would disturb the elaborate schedule. Secondly, strict HEM should not be encouraged, because the actual distance between an individual pair of patch is totally impossible to estimate. It is unreasonable to fit such hard pairs without qualification. Finally, the balance between the total modulation weights of negative pairs and positive pairs is fatal for optimization. An overwhelmed ratio

This work was supported by the National Natural Science Foundation of China under Grant no. 61773295.

The authors are with the Electronic Information School, Wuhan University, Wuhan, 430072, China (e-mail: jyma2010@gmail.com, dyx_acuo@whu.edu.cn).

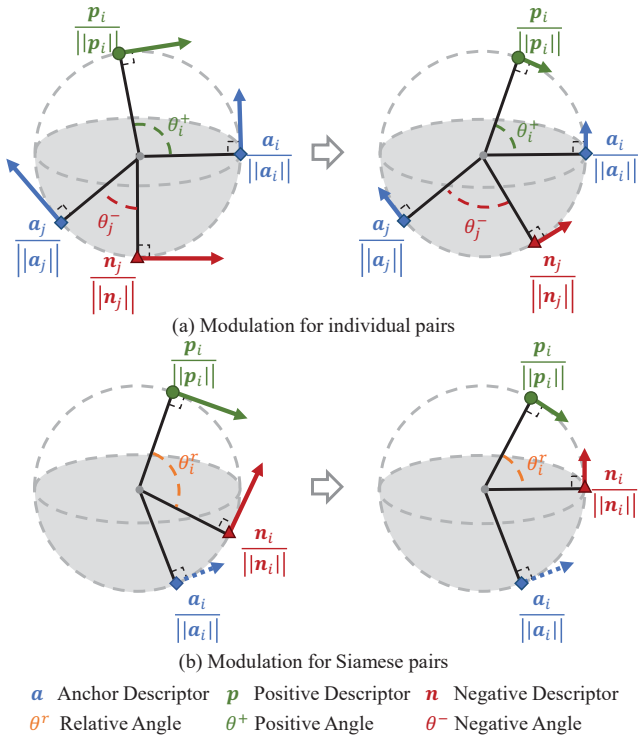


Fig. 1. Illustration of the necessity of gradient modulation. Arrows denote the gradients of descriptors during training. And subscripts represent the indices in a batch. Angle for a pair is defined in Eqn. (3). Relative angle θ_i^r is equal to $\theta_i^+ - \theta_i^-$. (a) The magnitude of the gradient of an individual matching pair $\{\boldsymbol{a}, \boldsymbol{p}\}$ should increase with θ^+. In contrast, the one of a non-matching pair $\{\boldsymbol{a}, \boldsymbol{n}\}$ should decrease with θ^-. (b) Siamese pair, *i.e.*, a triplet $\{\boldsymbol{a}, \boldsymbol{p}, \boldsymbol{n}\}$ in SDGMNet desires a smaller weight, when relative angle θ^r is diminishing.

of two total weights caused by the modulation would break down the training. In contrast, an appropriate ratio would help convergence and improve generalization.

In this paper, we propose SDGMNet, a statistic-based dynamic gradient modulation to tackle problems mentioned above. SDGMNet is also based on triplet loss proposed by HardNet with integrated four modifications. Firstly, we analyze the back-propagated gradient of triplet loss. Specifically, we explore that angular distance provides the relative flatten magnitude characteristic of gradients before modulation. SDGMNet is easily implemented by pair weighting with the included angle as distance measure (Section III-A). Secondly, we propose auto-focus modulation to modulate gradients for individual pairs. Auto-focus modulation utilizes the statistical characteristics of distances between individual pairs. Rather than following strict HEM principle, it mines statistically reliable pairs whose distances lay around the location of the distribution to orient the optimization (Section III-B). Thirdly, probabilistic margin employs statistical characteristics of relative distance of Siamese pairs, *i.e.*, triplets. It is applied to cut off the gradients of proportional Siamese pairs that are believed to reach the optimums. Meanwhile, the novel margin draws on hard mining for better convergence (Section III-C). Finally, we adjust the ratio of positive and negative total weights with weight normalization and attenuation coefficient (Section III-D). All statistics are estimated dynamically with rough Bayesian sequential update [3]. Extensive experiments on standard benchmarks including patch verification, matching, retrieval tasks confirm the superiority of the descriptors learned and fine-tuned in SDGMNet's formulation.

Our contributions can be summarized as follows:

1) We explore the special characteristic of angular distance in back propagation, which can provide a theoretical unbiased modulation for elaborated modifications.
2) We propose statistics-based auto-focus modulation to moderate the adverse impacts of the extreme situations carried by individual pairs so that the training can converge more stably.
3) We propose probabilistic margin, which combine CDF-based soft and hard margin, to mine and further optimize those hardest Siamese pairs in a more delicate and explainable way.
4) We propose power adjustment to rebalance total weights of negative and positive pairs for targeted learning and better generalization.

II. Related Works

In current years, modulating gradients becomes the shared theme of designing pair-based loss functions in metric learning. Modulation strategies can be categorized into two classes: Modulation for individual pairs and modulation for Siamese pairs. We briefly illustrate these two cases with triplet-based loss as an example in Fig. 1. Our method also stands on them. Thus, we try to absorb related modulation strategies no matter whether they are dynamic or exclusive for triplet loss. For better analyzing these cases in related works, let us represent the general loss as $\mathcal{L}$. $d^+(\boldsymbol{a}, \boldsymbol{p})$ denotes the general distance between a matching pair $\{\boldsymbol{a}, \boldsymbol{p}\}$, and $d^-(\boldsymbol{a}, \boldsymbol{n})$ denotes the one between non-matching $\{\boldsymbol{a}, \boldsymbol{n}\}$. For convenience, we simplify $d(\cdot)$ as d. $d^+ - d^-$ is usually referred to relative distance, denoted by d^r. θ is an instance for d. Note that, distance and negative similarity are not distinguished deliberately in this paper.

Modulation for Individual Pairs. Classical softmax loss has treated individual positive and negative pairs unequally. But it is not aware of that while $\partial\mathcal{L}/\partial d^+$ should be modulated with an increasing function in terms of d^+, $-\partial\mathcal{L}/\partial d^-$ needs a decreasing one, if we follow HEM. In recent years, Binomial deviance loss [57] utilizes softplus function to satisfy respective demands of d^+ and d^-. Circle loss [44] fulfills this purpose with circle margin. SFace [62] employs sigmoid functions to mine hard pairs. Meanwhile, SFace finds that the functions are disturbed in deep back propagation. But this problem is left alone and defended to restrain the noise in datasets, *e.g.*, MS-Celeb-1M [12]. For local descriptor learning, Keller *et al.* [17] follow HEM to modulate $\partial\mathcal{L}/\partial d_i^+$ and $-\partial\mathcal{L}/\partial d_i^-$ with functions symmetrical about $(d_i^+ + d_i^-)/2$ for each triplet. Then global information is crudely fused by shifting the axis of symmetry. Exp-TL [51] conducts more powerful HEM with exponential loss. The new loss makes $\partial\mathcal{L}/\partial d^+$ increase and $-\partial\mathcal{L}/\partial d^-$ decease exponentially. HyNet [46] also observes a hidden modulation in deep back propagation. It substitutes hybrid similarity for common similarity, so the hidden modulation is recast. The new modulation balances the needs of two kinds of pairs.

Modulation for Siamese Pairs. The relative hardness of Siamese pairs should be also taken into account. It provides more reliable information about the data distribution near the shared anchor. For a triplet, its hardness can be measured by d^r. Harder triplets with larger d^r should be emphasized with larger $\partial\mathcal{L}/\partial d^r$ during stochastic gradient descent. Balntas *et al*. [2] introduce a static hard margin for local descriptor learning. The hard margin modulates $\partial\mathcal{L}/\partial d^r$ with step function in terms of d^r and prevents easy triplets from backward propagation. Additionally, quadratic triplet loss in SOSNet [48] and scale-aware negative logarithmic softmax loss introduced by Keller *et al*. [17] polish original triplet loss with 'soft margin'. Thus, $\partial\mathcal{L}/\partial d^r$ is modulated by continuous functions that monotonically increase with d^r. Zhang and Rusinkiewicz [59] further enroll cumulative distribution function (CDF) to formulate a dynamic soft margin. Furthermore, n-pair losses are more popular with other metric learning tasks. In those cases, angular margin [20], [49], [6] cuts off partial easy Siamese pairs and achieves great success. Lifted structure loss [31] and multi-simi loss [53] separate Siamese pairs into independent positive and negative parts. The gradients of positive or negative pairs that share the same anchor are associated to be modulated. In contrast, circle loss [44] considers two kinds of Siamese pairs together.

Although not all of modulations discussed above are customized for triplet loss we try to modify, there is still something worthy of consideration. Especially, Zhang and Rusinkiewicz [59] have proposed a dynamic modulation for Siamese pairs with CDF, but it is not considerate. Moreover, the deep analysis on back-propagated gradients performed by HyNet [46] and SFace [62] suggests that the hidden modulation deserves more attentions. Additionally, the SFace points out that the strict HEM might be irrational, which indirectly explains the nature of hybrid similarity in HyNet. Adsorbing these ideas, we concentrate on our concerns and propose SDGMNet.

III. Methodology

Since SDGMNet majors in dynamically modulating gradients of pairs in triplet loss, a deep investigation on back-propagated gradients should be conducted. We define $\boldsymbol{x}(\boldsymbol{\Omega})$ and $\boldsymbol{y}(\boldsymbol{\Omega})$ as descriptors before normalization, where we omit the input patch and keep parameters of encoder $\boldsymbol{\Omega}$ as the only input. To facilitate subsequent analysis, we also discard $(\boldsymbol{\Omega})$. $\boldsymbol{a}$, $\boldsymbol{p}$ and $\boldsymbol{n}$ are instances of $\boldsymbol{x}$ or $\boldsymbol{y}$. Let N denote batch size, $\boldsymbol{D}$ denote the triplet distance batch $\{d_1^-, d_2^-, ..., d_N^-; d_1^+, d_2^+, ..., d_N^+\}$. Given a general function $f(\cdot)$ and a distance batch $\boldsymbol{D}$, the general triplet loss $\mathcal{L}$ can be represented as

$$\mathcal{L}(\boldsymbol{D}) = f(d_1^-, d_2^-, ..., d_N^-; d_1^+, d_2^+, ..., d_N^+). \tag{1}$$

The back-propagated gradient of the loss *w.r.t*. the parameters of the encoder $\boldsymbol{\Omega}$ can be computed with chain rule [11] as:

$$\begin{aligned}\frac{\partial\mathcal{L}}{\partial\boldsymbol{\Omega}} = &\sum_{i=1}^{N}\frac{\partial\mathcal{L}}{\partial d_i^+}\Big(\frac{\partial d_i^+}{\partial\boldsymbol{a}_i}\frac{\partial\boldsymbol{a}_i}{\partial\boldsymbol{\Omega}} + \frac{\partial d_i^+}{\partial\boldsymbol{p}_i}\frac{\partial\boldsymbol{p}_i}{\partial\boldsymbol{\Omega}}\Big) + \\ &\sum_{i=1}^{N}\frac{\partial\mathcal{L}}{\partial d_i^-}\Big(\frac{\partial d_i^-}{\partial\boldsymbol{a}_i}\frac{\partial\boldsymbol{a}_i}{\partial\boldsymbol{\Omega}} + \frac{\partial d_i^-}{\partial\boldsymbol{n}_i}\frac{\partial\boldsymbol{n}_i}{\partial\boldsymbol{\Omega}}\Big),\end{aligned} \tag{2}$$

where $(\boldsymbol{D})$ is omitted. In Eqn. (2), $\partial\mathcal{L}/\partial d$ is a scalar. It reveals how much the corresponding pair contributes to update of parameters. Gradient modulation for pairs focuses on rescaling $\partial\mathcal{L}/\partial d$ with a function about d. In this way, stochastic gradient descent optimization can be better controlled. However, related works [62], [46] explore that the term $\partial d/\partial\boldsymbol{x}$ harbors a hidden modulation function about d which would break our intention.

A. Angular Distance

Consider the term $\partial d/\partial\boldsymbol{x}$. Due to normalization, descriptors are embedded onto unit hypersphere. Included angular θ can be used for distance measure, defined as

$$\theta = \operatorname{acos}\frac{\boldsymbol{x}^T\boldsymbol{y}}{\|\boldsymbol{x}\|\|\boldsymbol{y}\|}, \tag{3}$$

where $\|\cdot\|$ denotes L_2 normalization. There are two more common metrics for distance measure: inner product s and L_2 distance l, which are defined as

$$s = \frac{\boldsymbol{x}^T\boldsymbol{y}}{\|\boldsymbol{x}\|\|\boldsymbol{y}\|}, \qquad l = \Big\|\frac{\boldsymbol{x}}{\|\boldsymbol{x}\|} - \frac{\boldsymbol{y}}{\|\boldsymbol{y}\|}\Big\|. \tag{4}$$

They are equivalent instances of d in forward propagation but distinguishing in back propagation. In back propagation, $\partial\theta/\partial\boldsymbol{x}$, $\partial s/\partial\boldsymbol{x}$ and $\partial l/\partial\boldsymbol{x}$ share the same optimal direction which is orthogonal to $\boldsymbol{x}$ as illustrated in Fig. 1. However, they own special magnitudes as

$$\Big\|\frac{\partial\theta}{\partial\boldsymbol{x}}\Big\| = \frac{1}{\|\boldsymbol{x}\|}, \tag{5}$$

$$\Big\|\frac{\partial s}{\partial\boldsymbol{x}}\Big\| = \frac{1}{\|\boldsymbol{x}\|}\sqrt{1-s^2}, \tag{6}$$

$$\Big\|\frac{\partial l}{\partial\boldsymbol{x}}\Big\| = \frac{1}{\|\boldsymbol{x}\|}\frac{\sqrt{4l^2-l^4}}{2l}. \tag{7}$$

As shown above, an implicit modulation takes effect after a metric is chosen. The magnitude of $\partial\theta/\partial\boldsymbol{x}$ depends on $\|\boldsymbol{x}\|$ only, which would not disturb the modulation function about θ we design later. Thus, θ is a suitable choice for our further intention. As for the term $1/\|\boldsymbol{x}\|$, we free it for two reasons. Firstly, related works [33], [7] observe that such natural scales can accelerate the training and better reflect the data variance. Moreover, $\boldsymbol{a}$, $\boldsymbol{p}$ and $\boldsymbol{n}$ interchange in every batch. So the impact of $1/\|\boldsymbol{x}\|$ holds balanced globally.

In short, $\partial\theta/\partial\boldsymbol{x}$ owns the optimal direction and a plain magnitude for learning. Thus, we employ θ for distance measure in SDGMNet. As a result, we can dedicate to gradient modulation for pairs, *i.e*., formulating $\partial\mathcal{L}/\partial d$. Eqn. (2) can be reformulated with θ into

$$\frac{\partial\mathcal{L}}{\partial\boldsymbol{\Omega}} = \sum_{i=1}^{N} w_i^+\frac{\partial\theta_i^+}{\partial\boldsymbol{\Omega}} - \sum_{i=1}^{N} w_i^-\frac{\partial\theta_i^-}{\partial\boldsymbol{\Omega}}, \tag{8}$$

where $\partial\mathcal{L}/\partial d_i$ is replaced by a weight w_i for convenience. We decompose w_i into $w_s \times w_c$ in SDGMNet. w_s and w_c will be introduced in the following subsections.

B. Auto-focus Modulation

w_s is the weight for modulating the gradients of individual pairs. w_s is refereed to self weight because it is exclusive for a pair.

Ideally, the angular distances of individual negative and the positive pairs reach their own optimums at π and 0, respectively. If following HEM, the gradient of an angle that is further away from its optimum should be weighted more largely. In other words, the gradients of matching pairs should be modulated with w_s^+ that is monotonously increasing *w.r.t.* θ^+, and the non-matching with monotonously decreasing w_s^-. However, naive distances θ^+ and θ^- are not dependable. Although the hardest positive pairs with large θ^+ are collected correctly, extreme distortions they carry would damage the convergence of stochastic gradient descent optimization. For the hardest negative pairs of patches, while the real distance between them cannot be evaluated, we simply push their descriptors away. Moreover, local feature learning is featured with open-set [10], few-shot [5] and large-scale [21], which means overfitting on training set is risky. Thus, extreme individual pairs should be treated more cautiously. Successes of HyNet [46] and Sface [62] also imply that excessive HEM on individual pairs should not be advocated.

To neutralize the HEM and extreme individual pairs suppression, we propose auto-focus modulation to formulate dynamic w_s^+ and w_s^- in SDGMNet as

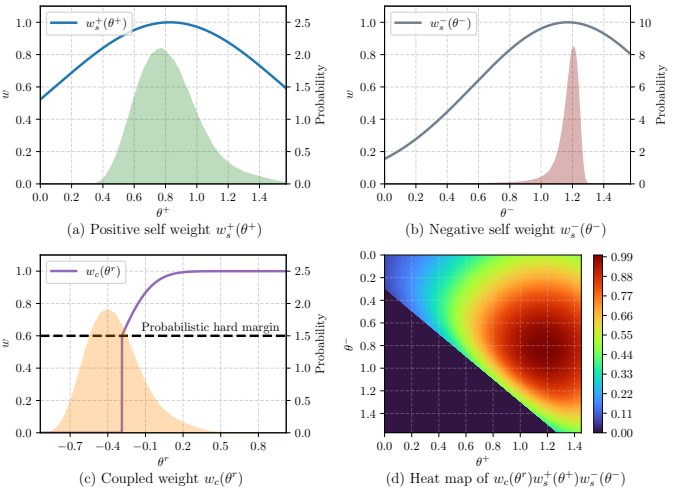
$$w_s^+(\theta^+) = \exp(-\frac{(\theta^+ - \mathrm{E}_t[\theta^+])^2}{2(\pi/6 + \mathrm{Std}_t[\theta^+])^2}), \tag{9}$$

$$w_s^-(\theta^-) = \exp(-\frac{(\theta^- - \mathrm{E}_t[\theta^-])^2}{2(\pi/6 + \mathrm{Std}_t[\theta^-])^2}), \tag{10}$$

where $\mathrm{E}_t[\cdot]$ represents the expectation, $\mathrm{Std}_t[\cdot]$ denotes the standard deviance, and the subscript t means the statistical characteristics are dynamically estimated over time. We visualize $w_s^+(\theta^+)$ and $w_s^-(\theta^-)$ at the last training iteration in Fig. 2 (a) and (b). The auto-focus modulation originates from Gaussian blur. It automatically focuses on the expectation of the positive or negative pairs that are more reliable examples we believe. Meanwhile, the impacts of harder and easier examples are weaken. It is worth mentioning that we limit the lower bound of the blur radius to $\pi/2$, *i.e.*, add $\pi/6$ to the standard deviance. If no constraint, the hardest examples that are long-tailed will be cleaned out with extremely small weight. Since the angle of positive pairs and negative pairs spread mainly on $[0, \pi/2]$ due to curse of dimensionality [60], a radius equal to $3(\pi/6 + \mathrm{Std}_t(\theta))$ is suitable to cover all examples with considerable weights.

C. Probabilistic Margin

The stochastic gradient descent optimization does not converge until $\partial\mathcal{L}/\partial\mathbf{\Omega}$ approaches zero. Thus, margins are necessary to force w in Eqn. (8) to be zeros near the optimums. For example, θ_i^+ holds an ideal optimum at 0, so the $w_i^+|_{\theta_i<0+m_i}$ should be 0, where the exclusive margin m_i can be an infinitesimal. However, it is impossible to search for m_i for each pair. Furthermore, two margins for matching and non-matching pairs are rational, but a joint margin set for triplets always proves more productive in practice. In other words, a single margin m designed as a function of θ^r is favored. The function modulates Siamese pairs with the same weight. The value of the function couples Siamese pairs, so we name it coupled weight w_c. How large m should be set for the global optimum is still unclear. Instead of employing a fixed empirical m, we intend to elaborate a hard margin $m \in [0, 1]$ that believes $100 \times m\%$ examples have reached the optimum, so-called probabilistic hard margin (PHM).



(a) Positive self weight $w_s^+(\theta^+)$ (b) Negative self weight $w_s^-(\theta^-)$ (c) Coupled weight $w_c(\theta^r)$ (d) Heat map of $w_c(\theta^r)w_s^+(\theta^+)w_s^-(\theta^-)$

Fig. 2. Visualization of modulation functions and related data distributions at the last training epoch on *Liberty*. Curves in (a), (b) and (c) illustrate three kinds of weights introduced in the text. Shadows denote the probability distribution of variables. (d) is a heat map of $w_s^+(\theta^+)w_s^-(\theta^-)w_c(\theta^r)$ with θ^+ and θ^- as x-axis and y-axis. The dark red in (d) indicates the strong impact on optimization. Our formulation does not turn the spotlight on hardest triplets that should lay at the top right corner. Easy triplets at the bottom left corner are eliminated by the hard margin.

Given a probabilistic hard margin m, we learn from the CDF-based soft margin [59] to form w_c in SDGMNet as

$$w_c(\theta^r) = \begin{cases} \mathrm{CDF}_t(\theta^r), & \theta^r > \mathrm{iCDF}_t(m), \\ 0, & \theta^r \le \mathrm{iCDF}_t(m), \end{cases} \tag{11}$$

where $\mathrm{CDF}(\cdot)$ denotes cumulative distribution function, $\mathrm{iCDF}(\cdot)$ denotes inverse cumulative distribution function. Triplets that carry θ^r smaller than $\mathrm{iCDF}(m)$ are top $100\times m\%$ easy examples empirically. These easy examples are believed to approach the optimum and will be isolated from further optimization. The others are preserved and weighted by a monotonously increasing CDF for HEM, since the relative distance of a Siamese pair is more reliable than the absolute distances of corresponding individual pairs. Due to probabilistic hardness, the modulation is dynamic and adaptive to training data and stage. To facilitate the implementation, we approximate the data distribution with a normal distribution. The curve of $w_c(\theta^r)$ is drawn in Fig. 2 (c), where we set $m = 0.6$. As can be seen, PHM steepens the CDF-based coupled weight to cut off proportional easy triplets.

So far, the coupled weight and self weight in SDGMNet are conceived completely.

Fig. 3. Network architecture adopted from HyNet [46]. Biases in convolution layers are activated except for the last one. Dropout regularization with 0.3 dropout rate is used before the last convolution layer.

D. Power Adjustment

Power is defined as the total weight of a class of pairs:

$$P^+ = \sum_{i=1}^{N} w_i^+, \qquad P^- = \sum_{i=1}^{N} w_i^-. \tag{12}$$

Power describes how strongly a class of positive or negative pairs guides the training with gradients. Before modulation, *i.e.*, $w = 1$, the positive power P^+ and the negative power P^- hold balanced. However, a bias has been introduced once the scale factors, *i.e.*, $1/\sqrt{2\pi}\sigma$ are dropped out in our formulation. Instead of determining proper scale factors in auto-focus modulation, we introduce power which involves modulations on both individual pairs and Siamese pairs to reconsider the bias. Intuitively, the positive power guides the model to identify the images with the same label. In contrast, the negative power forces the model to discriminate non-matching examples. An inductive bias on the negative power P^- would be preferred, because the model need not identify all labels well which would not appear in the test. Moreover, discriminating ability mutually promotes identifying for human beings. To adjust the ratio of power, we propose weight normalization that divides the weights by the expectation of the corresponding power. Then, attenuation is adopted on the positive side. Finally, SDGMNet is finished as:

$$\frac{\partial \mathcal{L}}{\partial \mathbf{\Omega}} = \frac{\alpha}{E_t[P^+]} \sum_{i=1}^{N} w_i^+ \frac{\partial \theta_i^+}{\partial \mathbf{\Omega}} - \frac{1}{E_t[P^-]} \sum_{i=1}^{N} w_i^- \frac{\partial \theta_i^-}{\partial \mathbf{\Omega}}, \tag{13}$$

where α is the attenuation coefficient. Once normalization functions, the ratio of the normalized powers can be quantified and adjusted by the attenuation coefficient. Such a ratio can endure random data, arbitrary modulations, running training phases and finally benefits the training.

E. Implementation

Triplet Sampling. Triplet sampling strategy proposed by HardNet [29] has become the de-facto standard for local descriptor learning. Briefly, HardNet follows L2Net [47] to sample N matching pairs. For a matching pair, HardNet mines the nearest negative neighbor as the negative example in the triplet. We follow the strategy and employ the anti-noise threshold for hard negative mining. The threshold in angular distance is set to 0.6.

Network Architecture. L2Net [47] proposes a classic encoder, in which batch normalization [15] and ReLU are employed after each convolution layer except for the last one.

Algorithm 1 SDGMNet for local descriptor learning

Input: SDGMNet hyperparameters m and α, vector of initial statistics $\boldsymbol{\beta}_0$, raw model, dataset, optimizer.

$t = 1$;
while training **do**
  Sample a data batch from datasets;
  Compute the angular distance matrix;
  Obtain HardNet triplets;
  Compute statistics $\boldsymbol{\mu}_t$ in batch, except for the expectation of powers;
  Update corresponding statistics by Eqn. (14);
  Compute $w_s^+(\theta^+)$, $w_s^-(\theta^-)$ and $w_c(\theta^r)$ by Eqns. (9), (10) and (11) respectively;
  if warming **then**
    Set $w_s^+(\theta^+)$, $w_s^-(\theta^-)$ and $w_c(\theta^r)$ to 1
  end if
  Compute powers P^+ and P^- by Eqn. (12);
  Update the expectation of powers by Eqn. (14);
  Construct pseudo loss by Eqn. (15);
  Call back propagation of pseudo loss;
  Update the model with the optimizer;
  $t = t + 1$;
end while

Output: Well-trained model.

HyNet [46] replaces these normalization layers to learnable Filter Response Normalization (FRN) and TLU [43]. Moreover, it introduces an additional normalization layer at the input of the encoder. These modifications remarkably improve local deep descriptor learning with little cost. We adopt this network architecture and illustrate it in Fig. 3.

Statistics Estimation. There are some statistical arguments varying over time in SDGMNet, which make the modulation dynamic. We employ rough Bayesian sequential update [3] to estimate these arguments as:

$$\boldsymbol{\beta}_t = 0.999\boldsymbol{\beta}_{t-1} + 0.001\boldsymbol{\mu}_t, \tag{14}$$

where $\boldsymbol{\beta}_t$ is the vector of approximated global statistics and $\boldsymbol{\mu}_t$ is the estimation in batch at the tth iteration. The past data provide a priori to current training. A fixed replacement rate 0.001 is sensitive enough for the former stage and stable for the latter stage of training.

Pseudo Loss. The modulated gradient in SDGMNet contains CDF that is a non-elementary function so that we cannot find a simple direct loss to guide the training. Motivated by general pair weighting framework [53], we define pseudo loss as:

$$\mathcal{L}_{\mathcal{P}} = \frac{\alpha}{E_t[P^+]} \sum_{i=1}^{N} w_i^+ \theta_i^+ - \frac{1}{E_t[P^-]} \sum_{i=1}^{N} w_i^- \theta_i^-, \tag{15}$$

where α, w, $E_t[P^+]$ and $E_t[P^-]$ are all constant with regard to the model parameter $\mathbf{\Omega}$. The gradient of pseudo loss is the same as Eqn. (13) so it can be used to exercise the SDGMNet. Specifically, except for θ, variables in pseudo loss are calculated firstly at each iteration and detached from back propagation. Then calling backward function of pseudo

TABLE I
PATCH VERIFICATION PERFORMANCE ON UBC PHOTOTOUR. NUMBERS SHOWN ARE FPR@95 (%) THAT ARE LOWER FOR BETTER. THE **BEST** SCORES ARE HIGHLIGHTED IN BOLD. DASH LINES INDICATE CHANGES OF MODELS. LIB: *Liberty*, YOS: *Yosemite*, ND: *Notredame*.

Train	ND	YOS	LIB	YOS	LIB	ND	Mean
Test	LIB		ND		YOS		
SIFT [22]	29.84		22.53		27.29		26.55
TFeat [2]	7.39	10.13	3.06	3.80	8.06	7.24	6.64
L2Net [47]	2.36	4.70	0.72	1.29	2.57	1.17	2.23
HardNet [28]	1.49	2.51	0.53	0.78	1.96	1.84	1.51
CDFDesc [59]	1.21	2.01	0.39	0.68	1.51	1.29	1.38
SOSNet [48]	1.08	2.12	0.34	0.67	1.03	0.95	1.03
HN-FRN [28], [46]	1.26	1.76	0.41	0.58	1.16	1.05	1.04
HyNet [46]	0.89	**1.37**	**0.34**	0.61	0.88	0.96	0.84
SDGMNet	**0.88**	1.41	**0.34**	**0.46**	**0.82**	**0.69**	**0.78**

loss will yield gradients in Eqn. (13) for gradient descent optimization. Pseudo loss is a more general tool to guide the training with arbitrary gradient modulation strategies.

Training. We implement SDGMNet in PyTorch [32]. The procedure of SDGMNet is shown in Algorithm 1, where we set $m = 0.6$ and $\alpha = 0.9$ for the best performance. The network is trained for 200 epochs (200K iterations) with batch size of 1024 and SGD optimizer. Data augmentation is achieved by random rotation, flipping and cropping [28], [59], [41]. Momentum and weight decay of the optimizer are set to 0.9 and 0.0001, respectively. The learning rate is initialized with 1 and divided by 2 after each 10% of iterations. Moreover, the training is warmed up with $w = 1$ in the first 10% of iterations. During warming, only $E[P^+]$ and $E[P^-]$ function but all statistics are estimated in every iteration. As a result, only the initial values of $E[P^+]$ and $E[P^-]$ contribute to the full SDGMNet training. We set them to 10000 so that the learning can warm up more smoothly.

Fine-tuning. Undoubtedly, SDGMNet suffers from the shift of data distribution from the training to the test. Especially, the relative distance of a hardest triplet in the test, that contains a true correspondence and the nearest negative neighbor, cannot be properly estimated and optimized in training due to the data themselves and various batch processing, which leads to potential mismatching. To address the problem, the overall discrimination of features should be improved, *i.e.*, easier samples should be released for further optimization by relaxing the margin. To this end, we further fine-tune the pretrained model in 20 epochs with $m = 0.1$ and CDF-based probabilistic soft margin (PSM) deactivated.

IV. EXPERIMENTS

We experiment SDGMNet on four benchmarks: UBC PhotoTour [55], HPatches [1], Image Matching Challenge [16] and ETH 3D reconstruction [38]. The results are compared with state-of-the-art alternatives including SIFT [22], TFeat [2], L2Net [47], HardNet [28], CDFDesc [59], SOSNet [48] and HyNet [46]. All methods output 128-dimensional descriptors that can be evaluated with L_2 distance. Those learned with DNN are all trained with data augmentation. Note that, HardNet, CDFDesc and SOSNet enroll the same network architecture proposed by L2Net, which is deeper than TFeat's. HyNet further upgrades the network with FRN. We adopt the HyNet's architecture for SDGMNet and provide a version of HardNet equipped with FRN, named HN-FRN.

A. UBC PhotoTour

UBC PhotoTour [55] is the most widely used dataset for local descriptor learning. It consists of three subsets *Liberty*, *Yosemite* and *Notredame*. The whole dataset contains more than 1.5M patches and 500K labels. Deep descriptors are trained on one subset and tested on the other two. In the standard protocol, the test aims to verify 100K pairs of patches matching or not. We report the false positive rate at 95% recall (FPR@95) for verification results in Table I. Let A-B represent the result trained on A and then tested on B. Firstly, the gain of HyNet and SDGMNet on HN-FRN are considerable, which shows the efficiency of our theme, *i.e.*, gradient modulation on patch verification task. Compared with the HyNet, our SDGMNet obtains an improvement of 0.06 on the mean FPR@95. Specifically, SDGMNet outperforms HyNet on YOS-ND and ND-YOS with large margins of 0.15 and 0.27, when only on YOS-LIB a small gap exists. The better performance of SDGMNet proves the significance of making the gradient modulation dynamic.

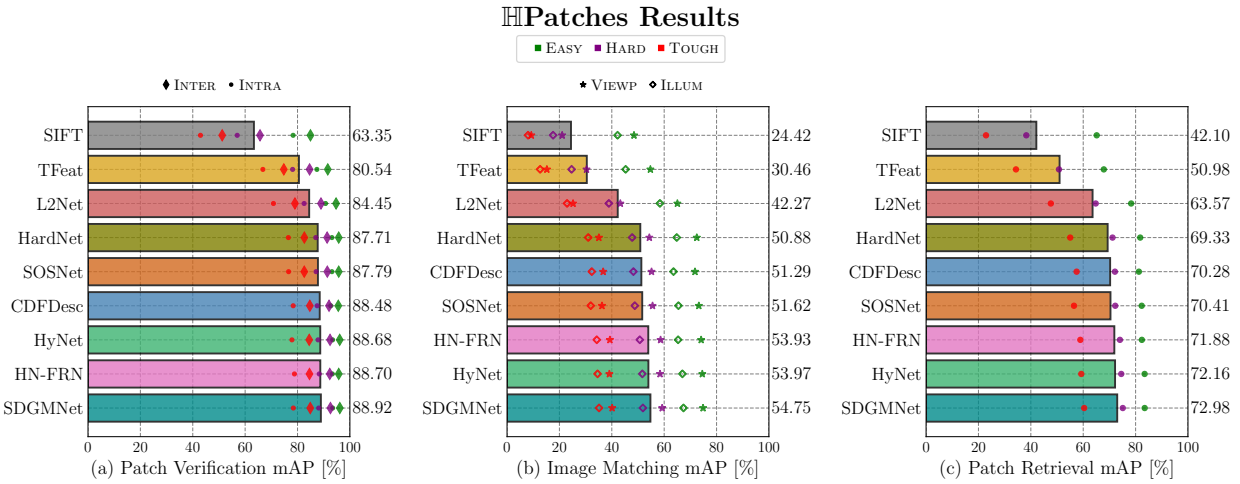
Moreover, since statistics are the cores of our dynamic modulation, we show all statistics on three subsets at several epochs in Tables II. As we can see, statistics varies among training phases and datasets. Especially, $\mathrm{E}[\theta^r]$ and $\mathrm{E}[\theta^+]$ vary apparently, which make the major contributions on our dynamic gradient modulation. In contrast, $\mathrm{E}[P^+]$ and $\mathrm{E}[P^-]$ keep stable. It suggests that modulation functions change with data distributions dynamically.

B. HPatches

HPatches [1] is a more comprehensive benchmark that evaluates descriptors on three tasks: patch verification, image matching and patch retrieval. According to geometric distortion, subtasks are categorized into *Easy*, *Hard* and *Tough*. Furthermore, patch pairs from the same or different image sequences are separated into two test subsets for verification, denoted by *Intra* and *Inter*, respectively. And the matching task is designed to evaluate the viewpoint (*VIEWP*) and illumination (*ILLUM*) invariance of descriptors. For a fair comparison,

TABLE II
STATISTICS IN FULL TRAINING PHASE ON *Liberty*| *Notredame*| *Yosemite*.

Epoch (-th)	30	70	110	150	190
E[θ^r] (10^{-1})	-2.86| -3.13| -3.61	-3.13| -3.40| -3.86	-3.30| -3.57| -4.00	-3.39| -3.65| -4.07	-3.43| -3.65| -4.11
Std[θ^r] (10^{-1})	2.37| 2.33| 2.46	2.31| 2.25| 2.41	2.25| 2.20| 2.36	2.21| 2.15| 2.33	2.18| 2.13| 2.31
E[θ^+] (10^{-1})	8.53| 8.26| 8.10	8.39| 8.14| 7.97	8.31| 8.06| 7.89	8.28| 8.02| 7.85	8.26| 8.01| 7.84
Std[θ^+] (10^{-1})	2.24| 2.18| 2.36	2.16| 2.09| 2.30	2.11| 2.01| 2.24	2.04| 1.96| 2.21	2.03| 1.95| 2.19
E[θ^-]	1.14| 1.14| 1.17	1.15| 1.15| 1.18	1.16| 1.16| 1.19	1.16| 1.17| 1.19	1.17| 1.17| 1.19
Std[θ^-] (10^{-2})	7.78| 8.66| 6.50	7.97| 8.60| 6.65	8.09| 8.60| 8.81	8.15| 8.60| 6.90	8.14| 8.60|6.97
E[P^+]	281| 281| 284	280| 281| 284	280| 281| 280	279| 281| 280	279| 281| 279
E[P^-]	299| 298| 304	297| 298| 301	295| 295| 300	294| 295| 298	294| 294| 297



(a) Patch Verification mAP [%] (b) Image Matching mAP [%] (c) Patch Retrieval mAP [%]

Fig. 4. Test on split 'a' of HPatches benchmark. All deep descriptors are trained on *Liberty* of PhotoTour. We report mean average precision (mAP) as evaluation metric. Results of subtasks are marked with different colors and patterns. The bars show the mean scores of subtasks. Mean scores are ranked from lowest to highest.

all models are trained on *Liberty* of UBC PhotoTour. As shown in Fig. 4, our SDGMNet surpasses predecessors on all three tasks. Especially, while there is no gap between HN-FRN and HyNet on image matching task, a remarkable improvement 0.82 on HN-FRN is achieved by SDGMNet. Moreover, SDGMNet exceeds HyNet with a margin 0.82 on patch retrieval task, which is larger than that between SOSNet and CDFDesc whose encoders are the same. These suggest that the dynamic modulation can help the descriptor learning and benefit the generalization of descriptors on various tasks.

To further evaluate descriptors on HPatches, we extract 1024 features per image, match them according nearest neighbor matching, and finally validate matches with ground-truth homography matrix. The visualization of matching results in Fig. 5 indicates the the superiority of our SDGMNet.

C. Image Matching Challenge

Although we achieve admirable results in UBC PhotoTour and Hpatches, the metrics are not evaluated under a standard matching pipeline which is the ultimate destination of descriptors. The rare inliers shown in Fig. 5 also prove that the metrics used above may not appropriately demonstrate the exact matching performances. Image Matching Challenge (IMC) [16] focuses on the performances of local features in

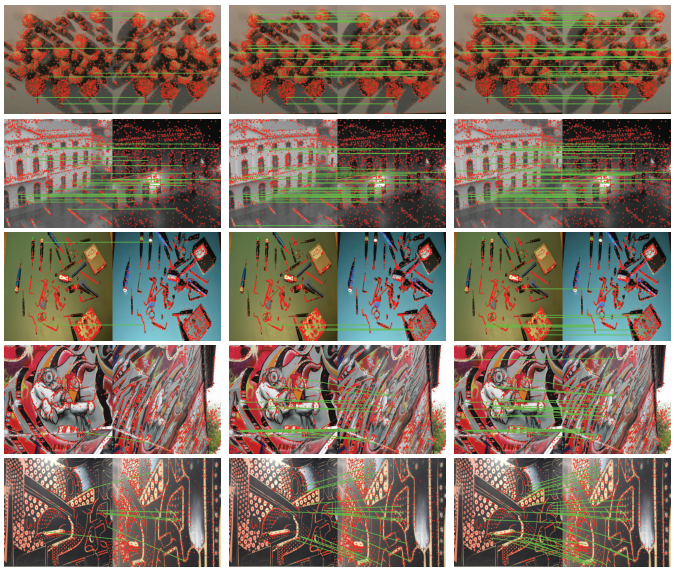

(a) SIFT (b) HardNet (c) SDGMNet

Fig. 5. Visualization of image matching performance on raw HPatches. The fist three rows of pairs suffer from illustration variance. The other pairs are pictured from different viewpoints. Keypoints are extracted by Difference-of-Gaussians (DOG) detector [22] and marked with '+' in pictures. The correctly matched pairs are linked with lines, while the others are not shown for clarity.

TABLE III
PERFORMANCES ON VALIDATION SET OF *PhotoTourism* IN IMAGE MATCHING CHALLENGE. EXCEPT FOR THE MULTIVIEW TASK WITH 8K KEYPOINTS TAKING RATIO TEST OF 0.8, THE OTHER TASKS SHARE THE SAME OPTIMAL RATIO TEST OF 0.9. THE **FIRST** AND THE **SECOND** BEST MAA (%) ARE MARKED IN COLOR.

#Keypoints	2k		8k		Mean
Task	Stereo	Multiview	Stereo	Multiview	
HardNet	54.32	61.64	67.07	74.24	64.32
CDFDesc	54.62	61.15	67.06	74.62	64.34
SOSNet	54.33	61.11	66.97	74.56	64.24
HyNet	54.71	**62.13**	67.29	**74.78**	**64.73**
HN-FRN	**55.23**	60.80	**68.10**	74.32	64.61
SDGMNet	**55.62**	**63.51**	**68.13**	**75.53**	**65.70**

both stereo and multiview pipelines with the accuracy of the reconstructed camera pose as the primary metric. It contains challenging scenes and reliable ground truths which make the benchmark more compelling. In the standard pipelines, we employ DOG as the detector and extract several state-of-the-art deep descriptors trained on *Liberty*. Then Fast Library for Approximate Nearest Neighbors (FLANN) is used to match the features for downstream tasks with the optimal ratio test. We use DEGENSAC [4] for geometric verification in stereo task with recommended settings. Please refer to [16] for more details.

The performances are shown in Table III with mean Average Accuracy to 10° angular error (mAA@10) as the evaluation metric. SDGMNet reaches state-of-the-art performance on all subtasks and finally obtains a gain about 1 on mean mAA, which demonstrates its practical applicability in a real image matching pipeline. We attribute the significant improvement to the special training strategy including full training and fine-tuning stages. More discussions are left in Section V-A.

D. ETH 3D Reconstruction

ETH benchmark [38] also quantifies descriptors on the image matching task. However, it shows more interest in how the matching performance affects the more challenging and practical 3D reconstruction tasks, *i.e.*, structure-from-motion (SFM) and Multi-View Stereo (MVS) [39], [37], [40]. To directly investigate the performance of descriptors, we abandon the ratio test in the standard pipeline of the benchmark. The performance of descriptors trained on *Liberty* are shown in Table IV. The number of registered images, reconstructed sparse points, dense points indicate the completeness of reconstruction. Mean track length and reprojection error are crucial for reconstruction accuracy. While none of descriptors is absolutely outstanding in relatively small subsets, SDGMNet achieves competitive scores in Madrid Metropolis and Gendarmenmarkt. Especially, SDGMNet generates the most complete model containing most sparse and dense points, which shows the superiority of SDGMNet in large 3D reconstruction tasks.

V. DISCUSSION

The advantages of SDGMNet have been clarified in comparative experiments above. To justify the impacts of components, hyperparameters and implementation details of SDGMNet, we conduct more elaborated study on several benchmarks. Because the full implementation of SDGMNet contains a full training stage and a fine-tuning stage, we would discuss the feasibility of fine-tuning at the first, and then analyze the other elements in full training stage for clarity.

A. Feasibility of Fine-tuning

SDGMNet without fine-tuning depends on the statistics estimated during training so that it is limited by the shift of data distribution among datasets and different batch processing. The weakness is expected to be magnified by the nearest neighbor matching principle. To compensate for the flaw, we fine-tune the pretrained model with a different m and PSM deactivated. Let SDGM-Raw represent the model trained without fine-tuning, SDGM-HPM represent the one fully trained with fine-tuning setting. The detailed performances of three training strategies are shown in Fig. 6 (a). FPR@95 of SDGM-Raw on *Liberty* can be found in Fig. 6 (c). In general, SDGMNet naturalizes outstanding performances in image matching and patch verification tasks, which are equally important for local descriptor learning.

Specifically, SDGMNet and SDGM-Raw surpass SDGM-HPM with certain margins in patch verification task, while m is varying in $(0, 0.6]$. SDGM-Raw with $m = 0.6$ are forced to learn on those hardest samples that are the bottleneck of this task in which the distance of pairs is relatively naturally distributed [60]. The ability is inherited to SDGMNet with a fine-tuning strategy so it keeps the competitive performance on *Liberty*. The similar tendency can be found on patch verification task of Hpatches.

In image matching task, the distribution of hardest triplets that determine matching performance is scenario-dependent. Thus, improving the overall discrimination of features should be the target rather than focusing on those 'hard hardest' samples. To this end, SDGMNet and SDGM-HPM pay more attention on easy samples with the slacken margin, which results in satisfied performances better than SDGM-Raw that shows almost no different due to PSM. Especially, SDGMNet significantly outperforms the other two at $m = 0.1$, which proves that fine-tuning can bring out a further improvement. The superior generalization is achieved probably because while SDGMNet is fine-tuned to 'review' easy samples, it has inherited the ability from SDGM-Raw to handle the hard ones.

TABLE IV
EVALUATION ON FIVE SUBSETS OF ETH 3D RECONSTRUCTION BENCHMARK. NUMBERS OF UNREGISTERED IMAGES IN DIFFERENT SUBSETS RANGE FROM 8 TO 1463. FIVE CRUCIAL METRICS FROM THE BENCHMARK ARE SELECTED TO REPORT. THE **BEST** AND THE **SECOND** SCORE ARE MARKED IN RED AND BLUE, RESPECTIVELY.

		#Reg. Images (↑)	#Sparse Points (↑)	#Dense Points (↑)	Track Length (↑)	Reproj. Error (↓)
Herzjesu (8 images)	HardNet	8	8.7K	238K	4.31	0.51px
	CDFDesc	8	8.7K	237K	4.31	0.52px
	SOSNet	8	8.7K	**239K**	4.31	**0.50px**
	HN-FRN	8	9.0K	238K	4.32	**0.50px**
	HyNet	8	8.9K	235K	4.31	0.53px
	SDGMNet	8	**9.0K**	237K	4.32	0.53px
Fountain (11 images)	HardNet	11	16.4K	296K	4.91	**0.47px**
	CDFDesc	11	16.4K	296K	4.92	0.48px
	SOSNet	11	16.4K	**297K**	4.91	**0.47px**
	HN-FRN	11	16.5K	**297K**	4.92	0.48px
	HyNet	11	16.5K	292K	4.92	0.49px
	SDGMNet	11	**16.6K**	296K	**4.94**	0.48px
South Building (128 images)	HardNet	128	161K	2.11M	**5.17**	0.63px
	CDFDesc	128	162K	2.10M	5.16	0.64px
	SOSNet	128	162K	2.09M	**5.17**	0.63px
	HN-FRN	128	165K	**2.12M**	5.16	0.64px
	HyNet	128	**166K**	**2.12M**	5.14	0.63px
	SDGMNet	128	**166K**	2.11M	5.16	0.65px
Madrid Metropolis (1334 images)	HardNet	723	258K	1.08M	3.72	0.96px
	CDFDesc	742	305K	1.23M	3.73	0.95px
	SOSNet	797	**393K**	**1.32M**	3.61	0.96px
	HN-FRN	**815**	384K	1.30M	3.57	0.97px
	HyNet	807	353K	1.25M	**3.83**	**0.94px**
	SDGMNet	**850**	**429K**	**1.35M**	**3.75**	**0.94px**
Gendarmenmarkt (1463 images)	HardNet	1163	795K	3.15M	3.79	1.03px
	CDFDesc	1145	783K	3.04M	**3.82**	**1.02px**
	SOSNet	1081	792K	2.68M	3.69	1.08px
	HN-FRN	**1204**	799K	**3.18M**	3.73	1.04px
	HyNet	1171	**836K**	3.13M	3.81	**1.01px**
	SDGMNet	**1184**	**841K**	**3.29M**	**3.90**	1.03px

Note that, FPR@95 of SDGMNet at $m = 0.1$ is 0.60 that is on par with 0.61 of HyNet [46], so m is set to 0.1 for fine-tuning.

In conclusion, although the fine-tuning seems a remedial action, it indeed helps SDGMNet attain better generalization, that SDGM-PHM cannot, which demonstrates the solid motivation of our formulation and implementation. Moreover, since SDGMNet relies on the strong initialization provided by SDGM-Raw, we mainly discuss how other elements take effect for SDGM-Raw on UBC PhotoTour, that is considered as the primary benchmark for local descriptor learning, in the subsequent sections.

B. Ablation Study

The raw formulation of SDGMNet, *i.e.*, SDGM-Raw contains four components including angular distance, auto-focus modulation (AF), probabilistic margin (PM) and power adjustment (PA). In fact, we think of angular distance from the perspective of gradient modulation for individual pairs, which shares the same motivation with AF. In other words, AF and implicit modulations in Eqns. (5), (6) and (7) are kinds of self weight, where the term $1/\|\boldsymbol{x}\|$ is omitted. Let $\&\theta$, $\&s$, $\&l_2$ denote self weights computed by Eqns. (5), (6), (7), respectively. And &AF represents our formulation. We assess frameworks that combine four kinds of self weight with PM-based coupled weight. Moreover, to test the efficiency of PA, we embed PA to those raw frameworks. The performances on full UBC PhotoTour are shown in Fig. 6 (b).

Without PA, all frameworks outperform the HardNet [29] embedded with FRN (HN-FRN). Their scores have been floating near the previous record of HyNet. It is worth mentioning that $\&l_2$−PA is equivalent to the HN-FRN upgraded with PM, which brings a gain about 0.18. In comparison, a CDF-based soft margin, *i.e.*, CDFDesc [59], improves the HardNet by 0.13 as shown in Table I. While the performance is almost saturated, our novel PM still generates a larger improvement. However, angular distance and AF reveal only a little distinction without PA. AF is not so effective probably because a bias is introduced in AF formulation. The bias would mightily mislead the training and degenerate the performance. So an inductive bias is introduced by PA to fix the problem. After PA is equipped, all raw frameworks advance. In such circumstance, $\&\theta$+PA and &AF+PA show their superiority. Especially, &AF+PA builds up a big lead. These outcomes

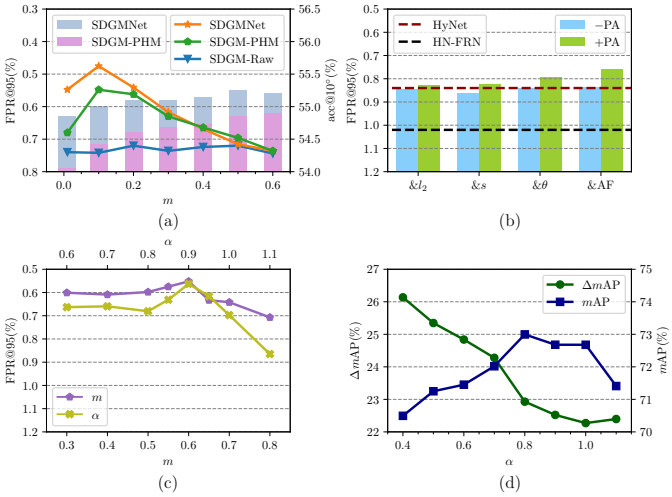


Fig. 6. Detailed experiments with different components, hyperparameters and implementation details. (a) Feasibility of fine-tuning. The bars indicate FPR@95 of SDGMNet and SDGM-PHM and with different m on *Liberty*. And the curves indicate the accuracy of our three training strategies on stereo task with 2k features of IMC. (b) Ablation experiments on full UBC PhotoTour. The red and black dashed line denote the mean FPR@95 of HyNet and HN-FRN, respectively. (c) Performances of raw SDGMNet on *Liberty* with different probabilistic margin m or attenuation coefficient α. (d) Impact of α on image retrieval task on HPatches. The last checkpoints on *Liberty* are selected to test on HPatches. While mAP is the mean on all subsets, ΔmAP denotes the gap between performances on Tough and Easy.

suggest the advantages of angular distance, AF and PA.

C. Impact of Hyperparameters

Batch Size. Batch size seems important for statistics estimation in SDGM-Raw. However, batch size also affects the hardest negative mining. To distinguish the impact on statistics estimation (SE) from that on hardest negative mining (HNM), we use partial samples in a batch to estimate the statistics with a fixed batch size of 1024. The results are labeled with SE. We also change the batch size and use the full batch to estimate statistics, which is labeled with HNM. As shown in Table V, a smaller sample size for statistics estimation causes only a tiny fluctuation because of rough Bayesian sequential update. And a small batch size strongly degrades the performance, which is consistent to the discussion in previous works [28], [48].

Attenuation Coefficient α. Attenuation coefficient α is one of the most crucial hyperparameters as analyzed before. To evaluate its impacts, we train SDGM-Raw on *Liberty* with other hyperparameters fixed. The best checkpoints are reported in Fig. 6 (c). Obviously, the curve peaks at $\alpha = 0.9$, while the performances with the other settings remarkably drop, especially, when α turns larger. That proves the importance of the ratio of power and significance of power adjustment. To further investigate the hyperparameter, we test the models with different α on image retrieval task of HPatches. The accuracy is shown in Fig. 6 (d), where mAP is the gap of mean accuracy between *Easy* and *Tough* splits. While the ΔmAP peaks near $\alpha = 0.9$ (better than HyNet), the ΔmAP tends to decline along with a increasing α. It can be explained that a ratio smaller than 1 leads to a preference on the negative part. The extreme variance of positive pairs cannot be learned by the model trained with a small ratio. Therefore, hard positive pairs are regarded as negative ones when the ratio is small. In conclusion, the best ratio depends on the difficulty level of the task. Since the difficulty level can be quantified in image matching task, power adjustment for local descriptor learning will be helpful in practice.

TABLE V
IMPACTS OF BATCH SIZE ON LIBERTY. THE AVERAGES OF TOP 10 FPR@95(%) ARE REPORTED.

	64	128	256	512	1024
SE	0.68	0.66	0.61	0.62	0.57
HNM	1.18	0.97	0.75	0.60	0.57

Probabilistic Hard Margin m. The basic impact of PHM m is also evaluated on *Liberty* as shown in Fig. 6 (c). The curve of FPR@95 versus m shares the similar tendency with α. Compared with α, varying m leads to smaller fluctuation. Thus, we do not show the joint impact of two hyperparameters in this paper. A larger m can isolate easier triplets from optimization more completely than PSM, so the improvement comes up and peaks at $m = 0.6$. But if only a few examples are released into optimization by an excessively large hard margin, *e.g.*, fewer than 30% with $m = 0.7$, the model would overfit on current batch and the performance would be weaken. It is worth noting that, compared to m larger than 0.6, a smaller m only leads to slight fluctuation on the performance. That is because the impact of easy triplets has been heavily reduced by PSM as illustrated in Fig. 2 (c), and that is why PSM would be deactivated for fine-tuning.

VI. CONCLUSION

In this paper, we propose a statistic-based dynamic gradient modulation for local descriptor learning, called SDGMNet. SDGMNet devotes to dynamically rescaling the gradients of pairs. Firstly, SDGMNet conducts deep analysis on back propagation and chooses included angle for distance measure. The angular distance is unbiased for every pair in theory. Secondly, auto-focus modulation is applied to modulate the gradients of individual pairs. It neutralizes the HEM and extreme example suppression according to statistical characteristics of individual pairs. Thirdly, SDGMNet enrolls statistic-based probabilistic margin to modulate the gradients of Siamese pairs, *i.e.*, triplets. It combines hard and soft HEM on triplets to help stochastic gradient descent optimization converge. Finally, total weights, *i.e.*, powers of two kinds of pairs are adjusted by gradient normalization and attenuation coefficient. SDGMNet fulfills the theme of modulating gradients dynamically with systematical analysis. Local descriptors learned and fine-tuned in SDGM strategy show superiority on various tasks and datasets. Every modification in SDGMNet proves efficient by extensive experiments. Furthermore, our success confirms that some extra problems, including hidden modulation in deep back propagation, balancing powers according to the difficulty of tasks and adjusting margin for practical image matching, deserve more attentions and further study.

REFERENCES

[1] Vassileios Balntas, Karel Lenc, Andrea Vedaldi, and Krystian Mikolajczyk. Hpatches: A benchmark and evaluation of handcrafted and learned local descriptors. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pages 5173–5182, 2017.

[2] Vassileios Balntas, Edgar Riba, Daniel Ponsa, and Krystian Mikolajczyk. Learning local feature descriptors with triplets and shallow convolutional neural networks. In *Proc. British Mach. Vis. Conf.*, pages 1–11, 2016.

[3] Christopher M Bishop. *Pattern recognition and machine learning*. springer, 2006.

[4] Ondrej Chum, Tomas Werner, and Jiri Matas. Two-view geometry estimation unaffected by a dominant plane. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pages 772–779, 2005.

[5] Debasmit Das and CS George Lee. A two-stage approach to few-shot learning for image recognition. *IEEE Trans. Image Process*, 29:3336–3350, 2019.

[6] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pages 4690–4699, 2019.

[7] Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superpoint: Self-supervised interest point detection and description. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pages 224–236, 2018.

[8] Mihai Dusmanu, Ignacio Rocco, Tomas Pajdla, Marc Pollefeys, Josef Sivic, Akihiko Torii, and Torsten Sattler. D2-net: A trainable cnn for joint description and detection of local features. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pages 8092–8101, 2019.

[9] Patrick Ebel, Anastasiia Mishchuk, Kwang Moo Yi, Pascal Fua, and Eduard Trulls. Beyond cartesian representations for local descriptors. In *Proc. IEEE Int. Conf. Comput. Vis.*, pages 253–262, 2019.

[10] Chuanxing Geng, Sheng-jun Huang, and Songcan Chen. Recent advances in open set recognition: A survey. *IEEE Trans. Pattern Anal. Mach. Intell.*, 43(10):3614–3631, 2021.

[11] Ian Goodfellow, Yoshua Bengio, Aaron Courville, and Yoshua Bengio. *Deep learning*. MIT Press Cambridge, 2016.

[12] Yandong Guo, Lei Zhang, Yuxiao Hu, Xiaodong He, and Jianfeng Gao. Ms-celeb-1m: A dataset and benchmark for large-scale face recognition. In *Proc. Europ. Conf. Comput. Vis.*, pages 87–102, 2016.

[13] Kun He, Yan Lu, and Stan Sclaroff. Local descriptors optimized for average precision. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pages 596–605, 2018.

[14] Yuge Huang, Yuhan Wang, Ying Tai, Xiaoming Liu, Pengcheng Shen, Shaoxin Li, Jilin Li, and Feiyue Huang. Curricularface: adaptive curriculum learning loss for deep face recognition. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pages 5901–5910, 2020.

[15] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *Proc. Int. Conf. Mach. Learn.*, pages 448–456, 2015.

[16] Yuhe Jin, Dmytro Mishkin, Anastasiia Mishchuk, Jiri Matas, Pascal Fua, Kwang Moo Yi, and Eduard Trulls. Image matching across wide baselines: From paper to practice. *Int. J. Comput. Vis.*, 129(2):517–547, 2021.

[17] Michel Keller, Zetao Chen, Fabiola Maffra, Patrik Schmuck, and Margarita Chli. Learning deep descriptors with scale-aware triplet networks. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pages 2762–2770, 2018.

[18] Vijay Kumar BG, Gustavo Carneiro, and Ian Reid. Learning local image descriptors with deep siamese and triplet convolutional networks by minimising global loss functions. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pages 5385–5394, 2016.

[19] Jiayuan Li, Qingwu Hu, and Mingyao Ai. Rift: Multi-modal image matching based on radiation-variation insensitive feature transform. *IEEE Trans. Image Process.*, 29:3296–3310, 2019.

[20] Weiyang Liu, Yandong Wen, Zhiding Yu, Ming Li, Bhiksha Raj, and Le Song. Sphereface: Deep hypersphere embedding for face recognition. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pages 212–220, 2017.

[21] Ziwei Liu, Zhongqi Miao, Xiaohang Zhan, Jiayun Wang, Boqing Gong, and Stella X Yu. Large-scale long-tailed recognition in an open world. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pages 2537–2546, 2019.

[22] David G Lowe. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vis.*, 60(2):91–110, 2004.

[23] Zixin Luo, Tianwei Shen, Lei Zhou, Siyu Zhu, Runze Zhang, Yao Yao, Tian Fang, and Long Quan. Geodesc: Learning local descriptors by integrating geometry constraints. In *Proc. Europ. Conf. Comput. Vis.*, pages 168–183, 2018.

[24] Zixin Luo, Lei Zhou, Xuyang Bai, Hongkai Chen, Jiahui Zhang, Yao Yao, Shiwei Li, Tian Fang, and Long Quan. Aslfeat: Learning local features of accurate shape and localization. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pages 6589–6598, 2020.

[25] Jiayi Ma, Xingyu Jiang, Aoxiang Fan, Junjun Jiang, and Junchi Yan. Image matching from handcrafted to deep features: A survey. *Int. J. Comput. Vis.*, 129(1):23–79, 2021.

[26] Jiayi Ma, Ji Zhao, Jinwen Tian, Alan L Yuille, and Zhuowen Tu. Robust point matching via vector field consensus. *IEEE Trans. Image Process.*, 23(4):1706–1721, 2014.

[27] Andrea Migliorati, Attilio Fiandrotti, Gianluca Francini, and Riccardo Leonardi. Learnable descriptors for visual search. *IEEE Trans. Image Process.*, 30:80–91, 2020.

[28] Anastasiya Mishchuk, Dmytro Mishkin, Filip Radenovic, and Jiri Matas. Working hard to know your neighbor's margins: Local descriptor learning loss. In *Adv. Neural Inf. Process. Syst.*, pages 4829–4840, 2017.

[29] Yair Movshovitz-Attias, Alexander Toshev, Thomas K Leung, Sergey Ioffe, and Saurabh Singh. No fuss distance metric learning using proxies. In *Proc. IEEE Int. Conf. Comput. Vis.*, pages 360–368, 2017.

[30] Kevin Musgrave, Serge Belongie, and Ser-Nam Lim. A metric learning reality check. In *Proc. Europ. Conf. Comput. Vis.*, pages 681–699, 2020.

[31] Hyun Oh Song, Yu Xiang, Stefanie Jegelka, and Silvio Savarese. Deep metric learning via lifted structured feature embedding. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pages 4004–4012, 2016.

[32] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. In *Adv. Neural Inf. Process. Syst.*, pages 8026–8037, 2019.

[33] Rajeev Ranjan, Carlos D Castillo, and Rama Chellappa. L2-constrained softmax loss for discriminative face verification. *arXiv preprint arXiv:1703.09507*, 2017.

[34] Jerome Revaud, Philippe Weinzaepfel, César De Souza, Noe Pion, Gabriela Csurka, Yohann Cabon, and Martin Humenberger. R2d2: repeatable and reliable detector and descriptor. *arXiv preprint arXiv:1906.06195*, 2019.

[35] Ethan Rublee, Vincent Rabaud, Kurt Konolige, and Gary Bradski. Orb: An efficient alternative to sift or surf. In *Proc. IEEE Int. Conf. Comput. Vis.*, pages 2564–2571, 2011.

[36] Torsten Sattler, Will Maddern, Carl Toft, Akihiko Torii, Lars Hammarstrand, Erik Stenborg, Daniel Safari, Masatoshi Okutomi, Marc Pollefeys, Josef Sivic, et al. Benchmarking 6dof outdoor visual localization in changing conditions. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pages 8601–8610, 2018.

[37] Johannes L Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pages 4104–4113, 2016.

[38] Johannes L Schonberger, Hans Hardmeier, Torsten Sattler, and Marc Pollefeys. Comparative evaluation of hand-crafted and learned local features. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pages 1482–1491, 2017.

[39] Johannes L Schonberger, Filip Radenovic, Ondrej Chum, and Jan-Michael Frahm. From single image query to detailed 3d reconstruction. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pages 5126–5134, 2015.

[40] Johannes L Schönberger, Enliang Zheng, Jan-Michael Frahm, and Marc Pollefeys. Pixelwise view selection for unstructured multi-view stereo. In *Proc. Europ. Conf. Comput. Vis.*, pages 501–518, 2016.

[41] Connor Shorten and Taghi M Khoshgoftaar. A survey on image data augmentation for deep learning. *J. Big Data*, 6(1):1–48, 2019.

[42] Edgar Simo-Serra, Eduard Trulls, Luis Ferraz, Iasonas Kokkinos, Pascal Fua, and Francesc Moreno-Noguer. Discriminative learning of deep convolutional feature point descriptors. In *Proc. IEEE Int. Conf. Comput. Vis.*, pages 118–126, 2015.

[43] Saurabh Singh and Shankar Krishnan. Filter response normalization layer: Eliminating batch dependence in the training of deep neural networks. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pages 11237–11246, 2020.

[44] Yifan Sun, Changmao Cheng, Yuhan Zhang, Chi Zhang, Liang Zheng, Zhongdao Wang, and Yichen Wei. Circle loss: A unified perspective of pair similarity optimization. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pages 6398–6407, 2020.

[45] Yurun Tian, Vassileios Balntas, Tony Ng, Axel Barroso-Laguna, Yiannis Demiris, and Krystian Mikolajczyk. D2d: Keypoint extraction with describe to detect approach. In *Proc. Asian Conf. Comput. Vis.*, pages 1–18, 2020.

[46] Yurun Tian, Axel Barroso Laguna, Tony Ng, Vassileios Balntas, and Krystian Mikolajczyk. Hynet: Learning local descriptor with hybrid similarity measure and triplet loss. In *Adv. Neural Inf. Process. Syst.*, pages 7401–7412, 2020.

[47] Yurun Tian, Bin Fan, and Fuchao Wu. L2-net: Deep learning of discriminative patch descriptor in euclidean space. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pages 661–669, 2017.

[48] Yurun Tian, Xin Yu, Bin Fan, Fuchao Wu, Huub Heijnen, and Vassileios Balntas. Sosnet: Second order similarity regularization for local descriptor learning. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pages 11016–11025, 2019.

[49] Hao Wang, Yitong Wang, Zheng Zhou, Xing Ji, Dihong Gong, Jingchao Zhou, Zhifeng Li, and Wei Liu. Cosface: Large margin cosine loss for deep face recognition. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pages 5265–5274, 2018.

[50] Jian Wang, Feng Zhou, Shilei Wen, Xiao Liu, and Yuanqing Lin. Deep metric learning with angular loss. In *Proc. IEEE Int. Conf. Comput. Vis.*, pages 2593–2601, 2017.

[51] Shuang Wang, Yanfeng Li, Xuefeng Liang, Dou Quan, Bowu Yang, Shaowei Wei, and Licheng Jiao. Better and faster: Exponential loss for image patch matching. In *Proc. IEEE Int. Conf. Comput. Vis.*, pages 4812–4821, 2019.

[52] Song Wang, Xin Guo, Yun Tie, Lin Qi, and Ling Guan. Deep local feature descriptor learning with dual hard batch construction. *IEEE Trans. Image Process.*, 29:9572–9583, 2020.

[53] Xun Wang, Xintong Han, Weilin Huang, Dengke Dong, and Matthew R Scott. Multi-similarity loss with general pair weighting for deep metric learning. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pages 5022–5030, 2019.

[54] Xing Wei, Yue Zhang, Yihong Gong, and Nanning Zheng. Kernelized subspace pooling for deep local descriptors. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pages 1867–1875, 2018.

[55] Simon AJ Winder and Matthew Brown. Learning local image descriptors. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pages 1–8, 2007.

[56] Alessio Xompero, Oswald Lanz, and Andrea Cavallaro. A spatio-temporal multi-scale binary descriptor. *IEEE Trans. Image Process.*, 29:4362–4375, 2020.

[57] Dong Yi, Zhen Lei, Shengcai Liao, and Stan Z Li. Deep metric learning for person re-identification. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pages 34–39, 2014.

[58] Kwang Moo Yi, Eduard Trulls, Vincent Lepetit, and Pascal Fua. Lift: Learned invariant feature transform. In *Proc. Europ. Conf. Comput. Vis.*, pages 467–483, 2016.

[59] Linguang Zhang and Szymon Rusinkiewicz. Learning local descriptors with a cdf-based dynamic soft margin. In *Proc. IEEE Int. Conf. Comput. Vis.*, pages 2969–2978, 2019.

[60] Xu Zhang, Felix X Yu, Sanjiv Kumar, and Shih-Fu Chang. Learning spread-out local feature descriptors. In *Proc. IEEE Int. Conf. Comput. Vis.*, pages 4595–4603, 2017.

[61] Zichao Zhang, Torsten Sattler, and Davide Scaramuzza. Reference pose generation for long-term visual localization via learned features and view synthesis. *Int. J. Comput. Vis.*, 129(4):821–844, 2021.

[62] Yaoyao Zhong, Weihong Deng, Jiani Hu, Dongyue Zhao, Xian Li, and Dongchao Wen. Sface: Sigmoid-constrained hypersphere loss for robust face recognition. *IEEE Trans. Image Process.*, 30:2587–2598, 2021.