Abstract

We introduce a simple modification of local image descriptors, such as SIFT, that improves matching performance by 43.09% on the Oxford image matching benchmark and is implementable in few lines of code. To put it in perspective, this is more than half of the improvement that SIFT provides over raw image intensities on the same datasets. The trick consists of pooling gradient orientations across different domain sizes, in addition to spatial locations, and yields a descriptor of the same dimension of the original, which we call DSP-SIFT. Domain-size pooling causes DSP-SIFT to outperform by 28.29% a Convolutional Neural Network, which in turn has been recently reported to outperform ordinary SIFT by 11.54%. This is despite the network being trained on millions of images and outputting a descriptor of considerably larger size. Domain-size pooling is counter-intuitive and contrary to the practice of scale selection as taught in scale-space theory, but has solid roots in classical sampling theory.

1 Introduction

Local image descriptors, such as SIFT and its variants [29], are engineered to reduce variability due to illumination and vantage point while retaining discriminative power. This facilitates local correspondence between different views of the same underlying scene. In a wide-baseline matching task on the Oxford benchmark [33, 32], nearest-neighbor SIFT descriptors achieve a mean average precision (mAP) of 27.50%, best among all local descriptors tested and a 71.85% improvement over direct comparison of grayscale values (16% mAP). Similar results have been obtained on other datasets [34]. Functions that reduce sensitivity to nuisance variability can also be learned from data [31, 42, 43, 48, 36], both supervised and unsupervised. Convolutional Neural Networks (CNNs) have been trained to “learn away” nuisance variability [17] while retaining class labels using large annotated datasets such as ImageNet [12, 24]. In particular, [17] use (patches of) natural images as surrogate classes per se, and add transformed versions of them to the training set while retaining class labels, thus training the network to discount nuisance variability. The response of a CNN to image values in a region can be interpreted as a descriptor, and used for correspondence much in the same way as SIFT, albeit with different dimension (the fourth layer of the CNN in [17, 14] has 8192 dimensions, whereas SIFT has 128). Direct comparison on the Oxford benchmark, performed by [17], shows that the CNN outperforms SIFT by a significant margin\(^1\) (11.54%), from 27.5% mAP to 30.68%. However, we show that a simple modification of SIFT, obtained by pooling gradient orientations across different domain sizes (“scales”), in addition to spatial locations, improves it by an even bigger margin (43.09%), to 39.36% mAP, outperforming the best CNN by 24.39% despite a far larger dimensionality of the latter. We call the resulting descriptor “domain-size pooled” SIFT, or DSP-SIFT for short.

Pooling across different domain sizes is implemented in few lines of code, can be applied to any histogram-based descriptor (Sect. 3), and yields a descriptor of the same size that outperforms the original essentially uniformly (i.e., on every single pair of images being matched, Fig. 3, third column). Yet, surprisingly, domain-size (DS) pooling has never been used in the design of local descriptors but for a few exceptions discussed next. This may be because it is

---

1These numbers are slightly different than those initially reported, following a notification by the authors of [17] that they found an issue in the interfacing with VLFeat that caused a glitch in their protocol for comparison to SIFT. The corrected protocol, following exchanges with the authors of VLFeat [45], is as follows: Select using MSER, but instead of mapping all selected regions to 91 × 91 pixels first, as done in [17], use the area of each selected and rectified region to determine the octave in scale-space where SIFT (as well as DSP-SIFT) is to be computed.

2This figure is slightly different from what is reported in [17] as we use a 50% intersection-over-union threshold rather than 60% (Sect. 4).
In practice we have found that the approximation has almost no minimum spacing of samples that will detect all extrema, a phenomenon indeed observed if pooling is not done properly. DS pooling also goes counter to the teachings of scale-space theory, and the resulting established practice of scale selection [28], as we discuss in Sect. 1.1. The rest of the paper is organized as follows: Sect. 2 describes what we do, Sect. 3 how we do it, and Sect. 5 why we do it. Section 4 validates our method empirically.

1.1 Related work

The “scale-invariant feature transform” SIFT [29] was designed to be robust to scale perturbations resulting from changes of viewpoint, exploiting the theory of scale-space and scale selection [28]. The SIFT descriptor can be written compactly as a formula, as explained in detail in [45]. A single, un-normalized cell of SIFT and its variants [11, 2, 8], can be written as a regularized histogram

\[
h_{\text{SIFT}}(\theta, \sigma) [x] = \int N_{\epsilon}(\theta - \angle\nabla I(y)) N_{\sigma}(y - x) d\mu(y), \quad x \in \Lambda(\sigma)
\]

where \( I \) is the image restricted to a square domain, centered at a location \( x \) with size \( \sigma \) determined by the response to a difference-of-Gaussian (DoG) operator across all locations and scales (SIFT detector). Here \( d\mu(y) = ||\nabla I(y)||dy, \theta \) is the independent variable, ranging from 0 to \( 2\pi \), corresponding to an orientation histogram bin of size \( \epsilon \), and \( \sigma \) is the spatial pooling scale. The kernel \( N_{\epsilon} \) is bilinear of size \( \epsilon \) and \( N_{\sigma} \) separable-bilinear of size \( \sigma \), although they could be replaced by a Gaussian with standard deviation \( \sigma \) and an angular Gaussian [47] with dispersion parameter \( \epsilon \). The SIFT descriptor is the concatenation of 16 cells (1) computed at locations \( x \in \{x_1, x_2, \ldots, x_{16}\} \) on a \( 4 \times 4 \) lattice \( \Lambda \), and normalized.

Note that the spatial pooling scale \( \sigma \) and the size of the image domain where the SIFT descriptor is computed \( \Lambda = \Lambda(\sigma) \) are tied to the photometric characteristics of the image, since \( \sigma \) is derived from the response of a DoG operator on the (single) image. Such a response depends on the reflectance properties of the scene and optical characteristics and resolution of the sensor, neither of which is related to the size and shape of co-visible (corresponding) regions. Instead, how large a portion of a scene is visible in each of two corresponding images depends on the shape of the scene, the pose of the two cameras, and the resulting visibility (occlusion) relations. Therefore, we propose to untie

---

3 Changing domain size brings into larger regions portions of the image that are not present in smaller regions, and rescaling causes the aggregation of sample gradient orientations from different portions of the scene, bringing together apples and oranges, which appears detrimental at first.

4 Last formula of http://www.vlfeat.org/api/sift.html.

5 Such a knot has its origins in harmonic analysis and wavelets, where it is referred to as “uncertainty principle” as it pertains to the task of reconstructing a single image from its compressed representation. Here the task is not reconstruction, but correspondence, where uncertainty is driven by geometry and topology (occlusion), rather than photometry (spatial frequencies).
the size of the domain where the descriptor is computed (“scale”) from photometric characteristics of the image, and do away with scale selection altogether. What should take its place to ensure robustness to scale changes, can be derived from first principles using classical sampling theory and anti-aliasing, which for histogram descriptors can be interpreted as averaging or “pooling” (Sect. 5).

Pooling is commonly understood as the combination of responses of feature detectors/descriptors at nearby locations, aimed at transforming the joint feature representation into a more usable one that preserves important information (intrinsic variability) while discarding irrelevant detail (nuisance variability) [4, 23]. However, precisely how pooling trades off these two conflicting aims is left vague and mostly addressed empirically in end-to-end comparisons with numerous confounding factors. Exceptions include [4], where intrinsic and nuisance variability are combined and abstracted into the variance and distance between the means of scalar random variables in a binary classification task. For more general settings, the goals of reducing nuisance variability while preserving intrinsic variability is elusive as a single image does not afford the ability to separate the two.

Our alternate interpretation of pooling as anti-aliasing clearly highlights its effects on intrinsic and nuisance variability: Because one cannot know what portion of an object or scene will be visible in a test image, a scale-space (“semi-orbit”) of domain sizes (“receptive fields”) should be marginalized. Proper marginalization cannot be computed in closed-form, so the semi-orbit has to be sampled. To reduce complexity, only a small number of samples should be retained, resulting in undersampling and aliasing phenomena. Anti-aliasing reduces such phenomena, with quantifiable effects on the sensitivity to nuisance variability. For the case of histogram-based descriptors, anti-aliasing planar translations consists of spatial pooling, routinely performed by most descriptors. Anti-aliasing translations along the optical axis results in domain-scale aggregation. This interpretation also offers a way to quantify the effects of pooling on discriminative (reconstruction) power directly, using classical results from sampling theory, rather than indirectly through an end-to-end classification experiment that may contain other confounding factors.

DS pooling can be applied to a number of different descriptors or convolutional architectures. We illustrate the effects of DS pooling on the most popular descriptor, SIFT. Importantly, optimal pooling requires the availability of multiple images of the same scene, and therefore cannot be performed in a single image. This fact has been lost in the design of local image descriptors, which are computed from a single image, with few exceptions [27, 44]. Of course, multiple images can be “hallucinated” from one, but the resulting pooling operation can only guarantee invariance for restricted classes of transformations [26]. This is also the case in our work.

In neural network architectures, there is evidence that abstracting spatial pooling hierarchically, i.e., aggregating descriptors that are nearby in feature space, is beneficial [4]. This process could be extended to DS-pooling, by aggregating across different neighborhood sizes in feature space. To the best of our knowledge, the only architecture that performs some kind of pooling across scales is [37], although the theoretical justifications provided in [5, 35] only concerns translation within each scale. The same goes for [6], where pooling (low-pass filtering) is only performed within each scale, and not across scales. Other works learn the regions for spatial pooling, for instance [23, 40], but still restrict pooling to within-scale, similar to [25], rather than across scales as we advocate. Interestingly, empirical studies [9] noticed no improvements by augmenting the data with scale transformations. As usual, with empirical comparisons the devil is lurking in the many parameters and confounding factors; on the other hand, our analysis (as well as experimental evidence) leads us to conclude otherwise.

We distinguish between multi-scale methods that concatenate descriptors computed independently at each scale, from cross-scale pooling, where statistics of the image at different scales are combined directly in the descriptor. Examples of the former include [22], where ordinary SIFT descriptors computed on domains of different size are assumed to belong to a linear subspace estimated with PCA, and [40], where Fisher vectors are computed for multiple sizes and aspect ratios and spatial pooling occurs within each level.

Aggregation across multiple views can be thought of as a form of temporal-pooling, to be performed along with domain-scale pooling. This appears to be championed in [21], but in the absence of precise definitions and formulas it is difficult to ascertain how this relates to our work. In [15], weights are shared across scales, which is not equivalent to pooling, but at least establishes some dependencies across scales.

To the best of our knowledge, pooling across scales is only done in HMAX for convolutional architectures [37], and in MTD for engineered descriptors [26]. Most recently, [19] advocates pooling across scales, but in practice space-pooled VLAD descriptors obtained at different scales are simply concatenated. Also Geometric Blur [3] can be thought of as a form of pooling, but the resulting descriptor only captures the mean of the resulting distribution.
Additional discussion in relation to details of prior related work is reported in Sect. B.2.

2 Contributions

If SIFT is written as (1), then DSP-SIFT is given by

$$h_{\text{DSP}}(\theta | I)[x] = \int h_{\text{SIFT}}(\theta | I, \sigma)[x] E_\sigma(\sigma) d\sigma \quad x \in \Lambda$$

(2)

where \( s > 0 \) is the size-pooling scale and \( E \) is an exponential or other unilateral density function. This is our main contribution. The process is visualized in Fig. 1. Unlike SIFT, that is computed on a scale-selected lattice \( \Lambda(\hat{\sigma}) \), DSP-SIFT is computed on a regularly sampled lattice \( \Lambda \). Computed on a different lattice, the above can be considered as a recipe for DSP-HOG [11]. Computed on a tree, it can be used to extend deformable-parts models (DPM) [16] to DSP-DPM. Replacing \( h_{\text{SIFT}} \) with other histogram-based descriptor “X” (for instance, SURF [2]), the above yields DSP-X. Applied to a hidden layer of a convolutional network, it yields a DSP-CNN, or DSP-Deep-Fisher-Network [39]. The details of the implementation are in Sect. 3.

While the implementation of DS pooling is straightforward, its justification is less so. We report the summary highlights in Sect. 5, that represent contributions to the understanding of pooling and the design and learning of local descriptors. The detailed derivation is described in Sect. B. It provides a theoretical justification for DS pooling and explicit conditions under which the resulting descriptors are valid. Nevertheless, one cannot forgo empirical validation on real images, where such conditions are routinely violated. In Sect. 4 we compare DSP-SIFT to alternate approaches. Motivated by the experiments of [33, 34] that compare local descriptors on wide-baseline matching benchmarks and show SIFT a clear winner, we choose SIFT as a paragon and compare it to DSP-SIFT on the standard benchmark [33]. Motivated by [17] that compares SIFT to both supervised and unsupervised CNNs trained on Imagenet and Flickr respectively, with the latter emerging as the clear winner on the same benchmark [33], we submit DSP-SIFT to the same evaluation protocol. We also run the test on the new synthetic dataset introduce by [17], that yields the same qualitative assessment. It should be noted that the comparison is unfair in favor of the CNNs, due to its increased dimension compared to SIFT and DSP-SIFT. Moreover, the best performance of a CNN is obtained using its fourth layer responses, that contain 8192 coefficients, a 64-fold complexity increase, even without accounting for the cost of learning, which is none for DSP-SIFT.

Clearly, DS pooling of under-sampled semi-orbits cannot outperform fine sampling, so if we were to retain all the scale samples instead of aggregating them, performance would further improve. However, computing a large collection of SIFT descriptors across different scales would incur significantly increased computational and storage cost. To contain the latter, [22] assume that descriptors at different scales populate a linear subspace and fit a high-dimensional hyperplane. The resulting Scale-less SIFT (SLS) outperforms ordinary SIFT as shown in Fig. 5. However, the linear subspace assumption breaks when considering large scale changes, so SLS is outperformed by DSP-SIFT despite the considerable difference in (memory and time) complexity.

3 Implementation and Parameters

Following common practice in evaluation protocols, we use maximally-stable extremal regions (MSER) [30] to detect candidate regions, affine-normalize them, align them to the dominant orientation, and re-scale them for comparison with [17]. For a detected scale \( \hat{\sigma} \), DSP-SIFT, align samples within a neighborhood \((\lambda_1 \hat{\sigma}, \lambda_2 \hat{\sigma})\) around it. For each scale-sampled patch, a single-scale un-normalized SIFT descriptor (1) is computed on the SIFT scale-space octave corresponding to the detected scale. By choosing \( E_\sigma \) to be a uniform density, these raw histograms of gradient orientations at different scales are accumulated and normalized\(^6\) to produce DSP-SIFT (2), which is compared to several descriptors. In the following evaluation, we use \( \lambda_1 = 1/6, \lambda_2 = 4/3 \) and \( N_{\hat{\sigma}} = 15 \). These parameters are empirically selected on the Oxford dataset [32, 33]. Fig. 4(a) shows that mean average precision (defined in Sect. 4.3) changes over the scale pooling range. An immediate advantage of DS pooling is observed when more than one scale

\(^6\)We follow the practice of SIFT [29] to normalize, clamp and re-normalize the histograms to make them more robust to contrast changes. The clamping threshold is set to 0.067 empirically.
Average Precision for different magnitude of transformations. The left 9 panels show (AP) for increasing magnitude of the 8 transformations in the Oxford dataset [32]. The mean AP over all pairs with corresponding amount of transformation are shown in the middle of the third row. The right 6 panels show the same for Fischer’s dataset [17].

is used. On the other hand, one should note that pooling over all scales as done in MTD [26] decreases performance as shown in the figure when the DS pooling radius exceeds 5/6. Fig. 4(b) shows the impact of the number of samples used to construct DSP-SIFT. It is not surprising that sampling more scales within the fixed DS pooling range improves performance over single-scale SIFT. Actually, the mAP of DSP-SIFT surpasses the ordinary SIFT even with 3 scales sampled. The performance gain saturates when more than 10 samples are used. Additional samples do not further increase the mean average precision, but incur more computational cost.

4 Validation

4.1 Paragons

As a baseline, the RAW-PATCH (a misnomer, following [17]) descriptor is the unit-norm grayscale intensity of the rectified and resized patch (91 × 91). Standard SIFT, which is widely accepted as a paragon [32, 34], is computed using VLFeat [45]. Both SIFT and DSP-SIFT descriptors are computed on the SIFT scale-space corresponding to the detected scales. Scale-less SIFT (SLS) is computed using the source code provided by the authors [22]: For each patch, standard SIFT is computed at 20 scales from $\sigma_{min} = 0.5$ to $\sigma_{max} = 12$, and the subspace dimension is set to 8 following [22], yielding a descriptor of dimension 8256. To compare DSP-SIFT to a convolutional neural network (CNN descriptor), as done by [17] for SIFT, we use their top-performing pre-trained unsupervised model on 16,000 natural images undergoing 150 transformations each (total 2.4M). Responses at layers 3 (CNN-L3) and 4 (CNN-L4) are used for comparison, following [17]. Since the network architecture requires the input to be of a fixed size, we tested and report the results on both 69 × 69 and 91 × 91 sizes used in [17]. Although no direct comparison with MTD [26] is performed, SLS can be considered as dominating it since it uses all scales without collapsing them into

---

7Following exchanges with the authors of VLFeat [45], instead of mapping all the patches to a user-defined size (e.g. 91 × 91), we use the area of each selected and rectified MSER region to determine the octave in the scale-space where SIFT (as well as DSP-SIFT) is to be computed.
SIFT defeats RAW−PATCH: 71.85%
AP with RAW−PATCH (4761d)
AP with SIFT (128d)

CNN−L4 defeats SIFT: 11.53%
AP with SIFT (512d)
AP with CNN−L4 (8192d)

DSP−SIFT defeats SLS: 4.63%
AP with SLS (8256d)
AP with DSP−SIFT (128d)

DSP−SIFT defeats CNN−L4: 28.29%
AP with CNN−L4 (8192d)
AP with DSP−SIFT (128d)

DSP−SIFT defeats SLS: 18.54%
AP with SLS (8256d)
AP with DSP−SIFT (128d)

DSP−SIFT defeats SIFT: 43.09%
AP with SIFT (128d)
AP with DSP−SIFT (128d)

DSP−SIFT defeats SLS: 18.53%
AP with SLS (8256d)
AP with DSP−SIFT (128d)

Figure 3: Head-to-head comparisons. Similarly to [17], each point represents one pair of images in the Oxford (top) and Fischer (bottom) datasets. The coordinates indicate average precision for each of the two methods under comparison. SIFT is superior to RAW-PATCH, but is outperformed by DSP-SIFT and CNN-L4. The right three columns show that DSP-SIFT is better than SLS and CNN-L4 despite the difference in dimensions (shown in the axes). The relative performance improvement of the winner is shown in the title of each panel.

a single histogram. The derivation in Sect. 5 suggests, and empirical evidence in Fig. 4(a) confirms, that aggregating the histogram across all scales significantly reduces discriminative power.

4.2 Datasets

The Oxford dataset [32, 33] comprises 40 pairs of images of mostly planar scenes seen under different pose, distance, blurring, compression and lighting. They are organized into 8 categories each of which consists of image pairs undergoing increasing magnitude of transformations. While routinely used to evaluate descriptors, this dataset has limitations in terms of size and restriction to mostly planar scenes, and modest scale changes. To alleviate these, Fischer et al. [17] recently introduced a dataset of 400 pairs of images with more extreme transformations than those in the Oxford dataset. The types of transformations include zooming, blurring, lighting change, rotation, perspective and nonlinear transformations. We use both datasets in our evaluation.

4.3 Metrics

Following [32], we use precision-recall (PR) curves to evaluate descriptors. A match between two detectors is called if their Euclidean distance is less than a threshold $d$. It is then labeled as a true positive if the area of intersection over union (IoU) of their corresponding MSER-detected regions is larger than 50%. Recall is defined as the fraction of true positives over the total number of correspondences. Precision is the percentage of true matches within the total number of matches. By varying the distance threshold $d$, a PR curve can be generated and the Average Precision (AP, a.k.a area under the curve, AUC) can be estimated. Average of APs provides the mean average precision (mAP) scores used for comparison.

---

Both datasets provide ground truth mapping between images, so the overlapping is computed by warping the first MSER region into the second image and then computing the overlap with the second MSER region.
Figure 4: Mean Average Precision for different parameters. (a) shows that mAP changes with the radius $s$ of DS pooling. The best mAP is achieved at $\hat{s} = 4/6$; (b) shows that mAP as a function of the number of samples used within the best range ($\hat{\sigma} - \hat{s}, \hat{\sigma} + \hat{s}$).

4.4 Comparison

Fig. 2 shows the detailed behavior (AP) of each descriptor as a function of the varying degree of severity of transformations. DSP-SIFT consistently outperforms other methods when there are scale changes (Zoom). It is also more robust to other transformations such as blur, lighting and compression in the Oxford dataset, [33], and to nonlinear, perspective, lighting, blur and rotation in Fischer’s [17]. DSP-SIFT is not at the top of the list of all compared descriptors in viewpoint change cases, although “viewpoint” is a misnomer as MSER-based rectification accounts for most of the viewpoint variability, and the residual variability is mostly due to interpolation and rectification artifacts. The fact that DSP-SIFT outperforms CNN in all cases in Fischer’s dataset is surprising, considering that the neural network is trained by augmenting the dataset using the similar types of transformations. Fig. 3 shows the head-to-head comparison between these methods, in the same format as [17]. Each point in the scatter plot represents one pair of images in both datasets. DSP-SIFT outperforms SIFT by 43.09% and 18.53% on Oxford and Fischer respectively. The performance gain comes without increase in dimension. In comparison, CNN-L4 achieves 11.54% and 11.53% improvements over SIFT by increasing dimension 64-fold. On both datasets, DSP-SIFT also consistently outperforms CNN-L4 and SLS by a large margin regardless of its lower dimension.

| Method    | Dim. | mAP Mikolajczyk | mAP Fischer |
|-----------|------|-----------------|-------------|
| SIFT      | 128  | .2750           | .4532       |
| DSP-SIFT  | 128  | .3936           | .5372       |
| CNN-L4-PS69 | 512 | .3059           | .4779       |
| CNN-L3-PS69 | 4096| .3164           | .4858       |
| RAW-PATCH | 4761 | .1600           | .3479       |
| CNN-L4   | 8192 | .3068           | .5055       |
| SLS       | 8256 | .3320           | .5135       |
| CNN-L3   | 9216 | .3056           | .4899       |

Table 1: Summary of complexity (dimension) and performance (mAP) for all descriptors sorted in order of increasing complexity. The lowest complexities and the best performances are highlighted in bold. We also report mAP for CNN descriptors computed on $69 \times 69$ patch to be consistent with [17].
4.5 Complexity and Performance Tradeoff

Fig. 5 shows the complexity (descriptor dimension) and performance (mAP) tradeoff for all descriptors considered. Table 1 summarizes the results on both datasets. In Fig. 5, an “ideal” descriptor would achieve mAP = 1 by using the smallest possible number of bits. So it will land in the top-left corner of the graph. DSP-SIFT has the same lowest complexity as SIFT and is the best in mAP among all the descriptors. Looking horizontally in the graph, DSP-SIFT outperforms all the other methods at a fraction of complexity. SLS achieves the second best performance but at the cost of a 64-fold increase in dimension. In general, the performance of CNN descriptors is worse than DSP-SIFT but, interestingly, their mAPs do not change significantly if the network responses are computed on a resampled patch of size $69 \times 69$ to obtain lower dimensional descriptors.

5 Derivation

In this section we describe the trace of the derivation of DSP-SIFT, which is reported in detail in Sect. B.
1. The likelihood function of the scene given images is a minimal sufficient statistic of the latter for the purpose of answering questions on the former [1]. Invariance to nuisance transformations induced by (semi-)group actions on the data can be achieved by representing orbits, which are maximal invariants [38]. The planar translation-scale group can be used as a crude first-order approximation of the action of the translation group in space (viewpoint changes) including scale change-inducing translations along the optical axis. This draconian assumption is implicit in most single-view descriptors.
2. Comparing (semi-)orbits entails a continuous search (non-convex optimization) that has to be discretized for implementation purposes. The orbits can be sampled adaptively, through the use of a co-variant detector and the associated invariant descriptor, or regularly - as customary in classical sampling theory.
3. In adaptive sampling, the detector should exhibit high sensitivity to nuisance transformations (e.g., small changes in scale should cause a large change in the response to the detector, thus providing accurate scale localization) and the descriptor should exhibit small sensitivity (so small errors in scale localization cause a small change in the descriptor). Unfortunately, for the case of SIFT (DoG detector and gradient orientation histogram descriptor), the converse is true (Fig. 6). This motivates us to resort to regular sampling.
4. Because correspondence entails search over samples of each orbit, reducing the number of samples is paramount. Undersampling introduces structural artifacts, or “aliasing,” corresponding to topological changes in the response of
Figure 6: Detector specificity vs. descriptor sensitivity. (Left) Change of detector response (red) as a function of scale, computed around the optimal location and scale (here corresponding to a value of 245), and corresponding change of descriptor value (blue). An ideal detector would have high specificity (sharp maximum around the true scale) and an ideal descriptor would have low sensitivity (broad minimum around the same). The opposite is true. This means that it is difficult to precisely select scale, and selection error results in large changes in the descriptor. Experiments are for the DoG detector and identity descriptor. Referring to the notation in Appendix (see details therein), (middle) template \( \rho \) (red) and target \( f \) (blue). (Right) corresponding scale-space \([f]\). Note that the maximum detector response may even not correspond to the true location. The jaggedness of the response is an aliasing artifact.

The jaggedness of the response can be reduced by “anti-aliasing,” an averaging operation (Fig. 7 right). For the case of (approximations of) the likelihood function, such as SIFT and its variants, anti-aliasing corresponds to pooling. While spatial pooling is common practice, and reduces sensitivity to translation parallel to the image plane, scale pooling – which would provide insensitivity to translation parallel to the optical axis – is not. This motivates the introduction of DSP-SIFT, and the rich theory on sampling and anti-aliasing could provide guidelines on what and how to pool, as well as bounds on the loss of discriminative power coming from undersampling and anti-aliasing operations.
Figure 7: Aliasing: (Top left) A random row is selected as the target $f$ and re-scaled to yield the orbit $[f]$; a subset of $f$, cropped, re-scaled, and perturbed with noise, is chosen as the template $\rho$. The distance $E$ between $\rho$ and $[f]$ is shown in red (right) as a function of scale. The same exercise is repeated for different sub-sampling of $[f]$, and rescaled for display either as a mesh (middle left) or heat map (right) that clearly show aliasing artifacts along the optimal ridge. Anti-aliasing scale (bottom) produces a cleaner ridge (left, right). The net effect of anti-aliasing has been to smooth the matching score $E$ (top-right, in blue) but without computing it on a fine grid. Note that the valley of the minimum is broader, denoting decreased sensitivity to scale, and the value is somewhat higher, denoting a decreased discriminative power and risk of aliasing if the value raises above that of other local minima.

6 Discussion

Image matching under changes of viewpoint, illumination and partial occlusions is framed as a hypothesis testing problem, which results in a non-convex optimization over continuous nuisance parameters. The need for efficient test-time performance has spawned an industry of engineered descriptors, which are computed locally so the effects of occlusions can be reduced to a binary classification (co-visible, or not). The best known is SIFT, which has been shown to “work well” in a number of independent empirical assessments [32, 34], that however come with little analysis on why it works, or indications on how to improve it. We have made a step in that direction, by showing that SIFT can
be derived from sampling considerations, where spatial binning and pooling are the result of anti-aliasing operations. However, SIFT and its variants only perform such operations for planar translations, whereas our interpretation calls for anti-aliasing domain-size as well. Doing so, surprisingly, can be accomplished in a few lines of code and yields significant performance improvements. Such improvements even place the resulting DSP-SIFT descriptor above a convolutional neural network (CNN), that had been recently reported as a top performer in the Oxford image matching benchmark [17]. Of course, we are not advocating replacing large neural networks with local descriptors, in general. We are just advocating properly designed descriptors in place of neural networks acting as such.

Domain-size pooling, and regular sampling of scale “unhinged” from the spatial frequencies of the signal is divorced from scale selection principles, rooted in scale-space theory, wavelets and harmonic analysis. There, the goal is to reconstruct a signal, with the focus on photometric nuisance factors (additive noise). In our case, the size of the domain where images correspond depends on the three-dimensional shape of the underlying scene, and visibility (occlusion) relations, and has nothing to do with the spatial frequencies or “appearance” of the scene. Thus, we do away with the linking of domain size and spatial frequency (“uncertainty principle”) to considerable performance benefits.

DS pooling can be easily extended to other descriptors, such as HOG, SURF, CHOG, including those supported on structured domains such as DPMs [16], and to convolutional networks, including Deep Fisher Networks [39], opening the door to multiple extensions of the present work. In addition, a number of interesting open theoretical questions can now be addressed using the tools of classical sampling theory, given the novel interpretation of SIFT and its variants introduced in this paper.

Acknowledgments

We are thankful to Nikolaos Karianakis for running the comparison against various forms of CNNs. This research was sponsored by ONR, ARO, AFOSR, NSF.

References

[1] R. R. Bahadur. Sufficiency and statistical decision functions. *Annals of Mathematical Statistics, 25*(3):423–462, 1954.

[2] H. Bay, T. Tuytelaars, and L. Van Gool. Surf: Speeded up robust features. *Computer Vision–ECCV 2006, pages 404–417, 2006.*

[3] A. Berg and J. Malik. Geometric blur for template matching. In *Proc. CVPR, 2001.*

[4] Y.-L. Boureau, J. Ponce, and Y. LeCun. A theoretical analysis of feature pooling in visual recognition. In *Proc. of Intl Conf. on Mach. Learning, pages 111–118, 2010.*

[5] J. V. Bouvrie, L. Rosasco, and T. Poggio. On invariance in hierarchical models. In *NIPS, pages 162–170, 2009.*

[6] J. Bruna and S. Mallat. Classification with scattering operators. In *Proc. IEEE Conf. on Comp. Vision and Pattern Recogn., 2011.*

[7] T.F. Chan and S. Esedoglu. Aspects of Total Variation Regularized L1 Function Approximation. *SIAM Journal on Applied Mathematics, 65*(5):1817, 2005.

[8] V. Chandrasekhar, G. Takacs, D. Chen, S. Tsai, R. Grzeszczuk, and B. Girod. Chog: Compressed histogram of gradients a low bit-rate feature descriptor. In *CVPR, pages 2504–2511, 2009.*

[9] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman. Return of the devil in the details: Delving deep into convolutional nets. *arXiv preprint arXiv:1405.3531, 2014.*

[10] C. Chen and H. Edelsbrunner. Diffusion runs low on persistence fast. In *ICCV, pages 423–430, 2011.*
[11] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *Proc. IEEE Conf. on Comp. Vision and Pattern Recogn.*, 2005. 2, 4

[12] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Computer Vision and Pattern Recognition, IEEE Conference on*, pages 248–255, 2009. 1

[13] J. Dong, J. Balzer, D. Davis, J. Hernandez, and S. Soatto. On the design and analysis of multiple view descriptors. *ArXiv: 1311.6048*, 2013. 15, 17, 18

[14] A. Dosovitskiy, J. T. Springenberg, and T. Brox. Unsupervised feature learning by augmenting single images. *arXiv:1312.5242*, 2013. 1

[15] C. Farabet, C. Couprie, L. Najman, and Y. LeCun. Scene parsing with multiscale feature learning, purity trees, and optimal covers. *arXiv preprint arXiv:1202.2160*, 2012. 3

[16] P. Felzenszwalb, D. McAllester, and D. Ramanan. A discriminatively trained, multiscale, deformable part model. In *CVPR*, pages 1–8, 2008. 4, 11, 18

[17] P. Fischer, A. Dosovitskiy, and T. Brox. Descriptor matching with convolutional neural networks: a comparison to sift. *ArXiv:1405.5769*, 2014. 1, 4, 5, 6, 7, 10, 17

[18] V. Fragoso, P. Sen, S. Rodriguez, and M. Turk. Evsac: Accelerating hypotheses generation by modeling matching scores with extreme value theory. In *ICCV*, pages 2472–2479, 2013. 16

[19] Y. Gong, L. Wang, R. Guo, and S. Lazebnik. Multi-scale orderless pooling of deep convolutional activation features. *arXiv preprint arXiv:1403.1840*, 2014. 3

[20] V. Guillemin and A. Pollack. *Differential Topology*. Prentice-Hall, 1974. 14

[21] P. Hamel, S. Lemieux, Y. Bengio, and D. Eck. Temporal pooling and multiscale learning for automatic annotation and ranking of music audio. In *ISMIR*, pages 729–734, 2011. 3

[22] V. Hassne, T. Mand Mayzels and L. Zelnik-Manor. On sifts and their scales. In *CVPR*, pages 1522–1528, 2012. 3, 4, 5

[23] Y. Jia, C. Huang, and T. Darrell. Beyond spatial pyramids: Receptive field learning for pooled image features. In *CVPR*, pages 3370–3377, 2012. 3

[24] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, pages 1097–1105, 2012. 1

[25] Y. LeCun. Learning invariant feature hierarchies. In *ECCV*, pages 496–505, 2012. 3

[26] T. Lee and S. Soatto. Learning and matching multiscale template descriptors for real-time detection, localization and tracking. In *Proc. IEEE Conf. on Comp. Vision and Pattern Recogn.*, pages 1457–1464, 2011. 3, 5

[27] T. Lee and S. Soatto. Video-based descriptors for object recognition. *Image and Vision Computing*, 2011. 3

[28] T. Lindeberg. Principles for automatic scale selection. Technical report, KTH, Stockholm, CVAP, 1998. 2

[29] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 2(60):91–110, 2004. 1, 2, 4, 16

[30] J. Matas, O. Chum, M. Urban, and T. Pajdla. Robust wide baseline stereo from maximally stable extremal regions. In *BMVC 2002*, 2002. 4

[31] R. Memisevic. Learning to relate images. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2013. 1

[32] K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. 2003. 1, 4, 5, 6, 10
[33] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. Van Gool. A comparison of affine region detectors. *IJCV*, 1(60):63–86, 2004. 1, 4, 6

[34] P. Moreels and P. Perona. Evaluation of features detectors and descriptors based on 3d objects. *International Journal of Computer Vision*, 73(3):263–284, 2007. 1, 4, 5, 10

[35] T. Poggio. How the ventral stream should work. Technical report, Nature Precedings, 2011. 3

[36] M. Ranzato, F. J. Huang, Y.-L. Boureau, and Y. LeCun. Unsupervised learning of invariant feature hierarchies with applications to object recognition. In *CVPR*, pages 1–8, 2007. 1

[37] T. Serre, A. Oliva, and T. Poggio. A feedforward architecture accounts for rapid categorization. *Proceedings of the National Academy of Sciences*, 104(15):6424–6429, 2007. 3

[38] J. Shao. *Mathematical Statistics*. Springer Verlag, 1998. 8

[39] K. Simonyan, A. Vedaldi, and A. Zisserman. Deep fisher networks for large-scale image classification. In *Advances in neural information processing systems*, pages 163–171, 2013. 4, 11

[40] K. Simonyan, A. Vedaldi, and A. Zisserman. Learning local feature descriptors using convex optimisation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2(4), 2014. 3

[41] G. Sundaramoorthi, P. Petersen, V. S. Varadarajan, and S. Soatto. On the set of images modulo viewpoint and contrast changes. In *Proc. IEEE Conf. on Comp. Vision and Pattern Recogn.*, June 2009. 18

[42] J. Susskind, R. Memisevic, G. E. Hinton, and M. Pollefeys. Modeling the joint density of two images under a variety of transformations. In *Proc. IEEE Conf. on Comp. Vision and Pattern Recogn.*, pages 2793–2800, 2011. 1

[43] G. W. Taylor, R. Fergus, Y. LeCun, and C. Bregler. Convolutional learning of spatio-temporal features. In *European Conference on Computer Vision*, pages 140–153. Springer, 2010. 1

[44] E. Tola, V. Lepetit, and P. Fua. Daisy: An efficient dense descriptor applied to wide-baseline stereo. *IEEE Trans. Pattern Anal. Mach. Intell.*, 32(5):815–830, 2010. 3

[45] A. Vedaldi and B. Fulkerson. Vlfeat: An open and portable library of computer vision algorithms. In *Proceedings of the international conference on Multimedia*, pages 1469–1472. ACM, 2010. 1, 2, 5

[46] C. Vondrick, A. Khosla, T. Malisiewicz, and A. Torralba. Hog-gles: Visualizing object detection features. In *Computer Vision (ICCV), 2013 IEEE International Conference on*, 2013. 18

[47] G. S. Watson. *Statistics on spheres*. Wiley, 1983. 2

[48] S. Winder and M. Brown. Learning local image descriptors. In *Proc. IEEE Conf. on Comp. Vision and Pattern Recogn.*, pages 1–8, 2007. 1
A Preliminaries

This first section summarizes the background needed for the derivation, reported in the next section.

A.1 Sampling and aliasing

In this section we refer to a general scalar signal \( f : \mathbb{R} \to \mathbb{R}; x \mapsto f(x) \), for instance the projection of the albedo of the scene onto a scanline. We define a detector to be a mechanism to select samples \( x_i \), and a descriptor \( \phi_i \) to be a statistic computed from the signal of interest and associated with the sample \( i \). In the simplest case, \( x \) is regularly sampled, so the detector does not depend on the signal, and the descriptor is simply the value of the function at the sample \( \phi_i = f(x_i) \). Other examples include:

A.1.1 Regular sampling (Shannon ’49)

The detector is trivial: \( \{x_i\} = \Lambda \) is a lattice, independent of \( f \). The descriptor is a weighted average of \( f \) in a neighborhood of fixed size \( \sigma \) (possibly unbounded) around \( x_i; \phi_i = \phi(\{f(x), x \in B_\sigma(x_i)\}) \). Neither the detector nor the descriptor function \( \phi \) depend on \( f \) (although the value of the latter, of course, does).

If the signal was band-limited, Shannon’s sampling theory would offer guarantees on the exact reconstruction \( \hat{f} \) of \( f(x), x \in \mathbb{R} \) from its sampled representation \( \{x_i, \phi_i\} \). Unfortunately, the signals of interest are not band-limited (images are discontinuous), and therefore the reconstruction \( \hat{f} \) can only approximate \( f \). Typically, the approximation includes ”alien structures,” i.e., spurious extrema and discontinuities in \( f \) that do not exist in \( f \). This phenomenon is known as aliasing. To reduce its effects, one can replace the original data \( f \) with another \( \hat{f} \) that is (closer to) bandlimited and yet close to \( f \), so that the samples can encode \( f = \hat{f} \) free of aliasing artifacts. The conflicting requirements of faithful approximation of \( f \) and restriction on bandwidth trade off discriminative power (reconstruction error) with complexity, which is one of the goals of communications engineering. This tradeoff can be optimized by choice of anti-aliasing operator, that is the function that produces \( \hat{f} \) from \( f \), usually via convolution with a low-pass filter. In our context, we seek for a tradeoff between discriminative power and sensitivity to nuisance factors. This will come naturally when anti-aliasing is performed with respect to the action of nuisance transformations.

A.1.2 Adaptive sampling (Landau ’67)

The detector could be “adapted” to \( f \) by designing a functional \( \psi \) that selects samples \( \{x_i\} = \psi(f) \). Typically, spatial frequencies of \( f \) modulate the length of the interval \( \delta x_i = x_{i+1} - x_i \). A special case of adaptive sampling that does not requires stationarity assumptions is described next. The descriptor may also depend on \( \psi \), e.g., by making the statistic depend on a neighborhood of variable size \( \sigma_i; \phi_i = \phi(\{f(x), x \in B_\sigma(x_i)\}) \).

A.1.3 Tailored sampling (Logan ’77)

For signals that are neither stationary nor band-limited, we can leverage on the violations of these assumptions to design a detector. For instance, if \( f \) contains discontinuities, the detector can place samples at discontinuous locations (”corners”). For band-limited signals, the detector can place samples at critical points (maxima, or “blobs”, minima, saddles). A (location-scale) co-variant detector is a functional \( \psi \) whose zero-level sets

\[
\psi(f; s, t) = 0
\]

(3)

define isolated (but typically multiple) samples of scales \( s_i > 0 \) and locations \( t_i \in \mathbb{R} \) locally as a function of \( f \) via the implicit function theorem [20], in such a way that if \( f \) is transformed, for instance via a linear operator depending on location \( \tau \) and scale \( \sigma \) parameters, \( W(\sigma, \tau)f \), then so are the samples: \( \psi(W(\sigma, \tau)f; s + \sigma, t + \tau) = 0 \).

The associated descriptor can then be any function of the image in the reference frame defined by the samples \( t_i, s_i \), the most trivial being the restriction of the original function \( f \) to the neighborhood \( B_{s_i}(t_i) \). This, however, does not reduce the dimensionality of the representation. Other descriptors can compute statistics of the signal in the neighborhood, or on the entire line. Note that descriptors \( \phi_i \) could have different dimensions for each \( i \).
A.1.4 Anti-aliasing and “pooling”

In classical sampling theory, anti-aliasing refers to low-pass filtering or smoothing that typically does not cause genetic phenomena (spurious extrema, or aliases, appearing in the reconstruction of the smoothed signal.) Of course, anti-aliasing typically has destructive effects, in the sense of eliminating extrema that are instead present in the original signal.

A side-effect of anti-aliasing, which has implications when the goal is not to reconstruct, but to detect or localize a signal, is to reduce the sensitivity of the relevant variable (descriptor) to variations of the samples (detector). If we sample translations, \( x_i = x + t_i \), and just store \( f_i = f(x_i) \), an arbitrarily small translation of the sample \( dx \) can cause an arbitrarily large variation in the representation \( \delta f(x_i) = f(x_i + dx) - f_i \), when \( x_i \) is a discontinuity. So, the sensitivity \( S(f) = \frac{\delta f}{dx} = \infty \). An anti-aliasing operator \( \phi(f) \) should reduce sensitivity to translation: \( \frac{\delta \phi(f)}{dx} \ll \frac{\delta f}{dx} \). Of course, this could be trivially achieved by choosing \( \phi(f) = 0 \) for any \( f \). The goal is to trade off sensitivity with discriminative power. For the case of translation, this tradeoff has been described in [6]. However, similar considerations holds for scale and domain-size sampling.

B Derivation of DSP-SIFT

The derivation of DSP-SIFT and its extensions follows a series of steps summarized as follows:

- We start from the correspondence, or matching, task: Classify a given datum \( f \) (test image, or target) as coming from one of \( M \) model classes, each represented by an image \( \rho_j \) (training images, or templates), with \( j = 1, \ldots, M \).

- Both training and testing data are affected by nuisance variability due to changes of (i) illumination (ii) vantage point and (iii) partial occlusion. The former is approximated by local contrast transformations (monotonic continuous changes of intensity values), a maximal invariant to which is the gradient orientation. Vantage point changes are decomposed as a translation parallel to the image plane, approximated by a planar translation of the image, and a translation orthogonal to it, approximated by a scaling of the image. Partial occlusions determine the shape of corresponding regions in training and test images, which are approximated by a given shape (say a circle, or square) of unknown size (scale). These are very crude approximations but nevertheless implicit to most local descriptors. In particular, camera rotations are not addressed in this work, although others have done so [13].

- Solving the (local) correspondence problem amounts to an \( M + 1 \)-hypothesis testing problem, including the background class. Nuisance (i) is eliminated at the outset by considering gradient orientation instead of image intensity. Dealing with nuisances (ii)–(iii) requires searching across all (continuous) translations, scales, and domain sizes.

- The resulting matching function must be discretized for implementation purposes. Since the matching cost is quadratic in the number of samples, sampling should be reduced to a minimum, which in general introduces artifacts (“aliasing”).

- Anti-aliasing operators can be used to reduce the effects of aliasing artifacts. For the case of (approximations of) the likelihood function, such as SIFT, anti-aliasing corresponds to marginalizing residual nuisance transformations, which in turn corresponds to pooling gradient orientations across different locations, scales and domain sizes.

B.1 Formalization

For simplicity, we formalize the matching problem for a scalar image (a scanline), and neglect contrast changes for now, focusing on the location-scale group and domain size instead.

---

9This central tenet of scale-space theory only holds for scalar signals. Nevertheless, genetic effects have been shown to be rare in two-dimensional Gaussian scale-space [10].
Let $\rho_j : \mathbb{R} \to \mathbb{R}$, with $j = 1, \ldots, M$ possible models (templates, or ideal training images). The data (test image) is $f : [0, \ldots, N] \to \mathbb{R}$ with each sample $f(x_i)$ obtained from one of the $\rho_j$ via translation by $\tau \in \mathbb{R}$, scaling by $\sigma > 0$, and sampling with interval $\epsilon$, if $x_i$ is in the visible domain $[a, b]$. Otherwise, the scene $\rho_j$ is occluded and $f(x_i)$ has nothing to do with it.

The forward model that, given $\rho$ and all nuisance factors $\sigma, \tau, a, b$, generates the data, is indicated as follows: If $x_i \in [a, b]$, then
\[
  f(x_i) = W_\epsilon(x_i; \sigma, \tau)\rho_j + n_{ij}
\]
where $n_{ij}$ is a sample of a white, zero-mean Gaussian random variable with variance $\kappa$. Otherwise, $x_i \notin [a, b]$, and $f(x_i) = \beta(x_i)$ is a realization of a process independent of $\rho_j$ (the “background”). The operator $W_\epsilon$ is linear\(^{10}\) and given by
\[
  W_\epsilon(x_i; \sigma, \tau) \rho \doteq \int_{B_\epsilon(x_i)} \rho \left( \frac{x - \tau}{\sigma} \right) dx
\]
where $B_\epsilon(x_i)$ is a region corresponding to a pixel centered at $x_i$. Matching then amount to a hypothesis testing problem on whether a given measured $f = \{f(x_i)\}_{i=1}^N$ is generated by any of the $\rho_j$ – under suitable choice of nuisance parameters – or otherwise is just labeled as background:
\[
  H_0 : \exists j, a, b, \sigma, \tau \mid p(f(x_i)|\rho_j, a, b, \sigma, \tau) = p_\beta(\{f(x_k), x_k \notin [a, b]\}) \prod_{x_i \in [a, b]} N(f(x_i) - W_\epsilon(x_i; \sigma, \tau)\rho_j, \kappa)
\]
and the alternate hypothesis is simply $p_\beta(\{f(x_i)\}_{i=1}^N)$. If the background density $p_\beta$ is unknown, the likelihood ratio test reduces to the comparison of the product on the right-hand side to a threshold, typically tuned to the ratio with the second-best match (although some recent work using extreme-value theory improves this\(^{18}\)). In any case, the log-likelihood for points in the interval $x_i \in [a, b]$ can be written as
\[
  r_{ij}(a, b, \sigma, \tau) = \frac{1}{|b - a|} \sum_{x_i \in [a, b]} |f(x_i) - W_\epsilon(x_i; \sigma, \tau)\rho_j|
\]
which will have to be minimized for all pixels $i = 1, \ldots, N$ and templates $j = 1, \ldots, M$, of which there is a finite number. However, it also has to be minimized over the continuous variables $a, b, \sigma, \tau$. Since $r$ is in general neither convex nor smooth as a function of these parameters, analytical solutions are not possible. Discretizing these variables is necessary,\(^{11}\) and since the minimization amounts to a search in $2 + 4$ dimensions, we seek for methods to reduce the number of samples with respect to the arguments $a, b, \sigma, \tau$ as much as possible.

There are many ways to sample, some described in Sect. A.1, so several questions are in order: (a) How should each variable be sampled? Regularly or adaptively? (b) If sampled regularly, when do aliasing phenomena occur? Can anti-aliasing be performed to reduce their effects? (c) The search is jointly over $a, b$ and $\sigma, \tau$, and given one pair, it is easy to optimize over the other. Can these two be “separated”? (d) Is it possible to quantify and optimize the tradeoff between the number of samples and classification performance? Or for a given number of samples develop the “best” anti-aliasing (“descriptor”)? (e) For a histogram descriptor, how is “anti-aliasing” accomplished?

### B.2 Common approaches and their rationale

Concerning question (a) above, most approaches in the literature perform tailored sampling (Sect. A.1.3) of both $\tau$ and $\sigma$, by deploying a location-scale covariant detector\(^{29}\). When time is not a factor, it is common to forgo
\[
\int_{B_\epsilon(x_i)} \rho \left( \frac{x - \tau}{\sigma} \right) dx = \int k_\epsilon(x - x_i)\rho \left( \frac{x - \tau}{\sigma} \right) dx = \int \delta \left( y - \frac{x - \tau}{\sigma} \right) k_\epsilon(x - x_i)\rho(y) dy dy = \int \delta \left( y + \frac{\tau}{\sigma} - x \right) k_\epsilon(\sigma y - x_i)\sigma dy dy = \sigma \int k_\epsilon(\sigma y + \tau - x_i)\rho(y) dy
\]
}\(^{11}\)Coarse-to-fine, homotopy-based methods or jump-diffusion processes can alleviate, but not remove, this burden.

\]
the detector and compute descriptors “densely” (a misnomer) by regularly subsampling the image lattice, or possibly undersampling by a fixed “stride.” Sometimes, scale is also regularly sampled, typically at far coarser granularity than the scale-space used for scale selection, for obvious computational reasons. In general, regular sampling requires assumptions on band limits. The function \( W \rho \) is not band-limited as a function of \( \tau \). Therefore, tailored sampling (detector/descriptor) is best suited for the translation group.\(^{12}\) We will therefore assume that \( \tau \) has been tailor-sampled (detected, or canonized), but only up to a localization error. Without loss of generality we assume the sample is centered at zero, and the residual translation \( \tau \) is in the neighborhood of the origin. In Fig. 6 we show that the sensitivity to scale of a common detector (DoG), which should be high, and is instead lower than the sensitivity of the resulting descriptor, which should be low. Therefore, small changes in scale cause large changes in scale sample localization, which in turn cause large changes in the value of the descriptor. Therefore, we forgo scale selection, and instead finely sample scale. This causes complexity issues, which prompt the need to sub-sample, and correspondingly to anti-alias or aggregate across scale samples. Alternatively, as done in Sect. 4, we can have a coarse adaptive or tailored sampling of scales, and then perform fine-scale sampling and anti-aliasing around the (multiple) selected scales.

Concerning (b), anti-aliasing phenomena appear as soon as Nyquist’s conditions are violated, which is almost always the case for scale and domain-size (Fig. 7). While most practitioners are reluctant to down-sample spatially, leaving millions of locations to test, it is rare for anyone to employ more than a few tens of scales, corresponding to a wild down-sampling of scale-space. This is true a fortiori for domain-size, where the domain size is often fixed, say to \( 69 \times 69 \) or \( 91 \times 91 \) locations [17]. And yet, spatial anti-aliasing is routinely performed in most descriptors, whereas none – to the best of our knowledge – perform scale or domain-size anti-aliasing. Anti-aliasing should decrease the sensitivity of the descriptor, without excessive loss of discriminative power. This is illustrated in Fig. 7.

For (c), we make the choice of fixing the domain size in the target (test) image, and regularly sampling scale and domain-size, re-mapping each to the domain size of the target (Fig. 1). For comparison with [17], we choose this to be \( 69 \times 69 \). While the choice of fixing one of the two domains entails a loss, it can be justified as follows: Clearly, the hypothesis cannot be tested independently on each datum \( f(x_i) \). However, testing on any subset of the “true inlier set” \([a, b]\) reduces the power, but not the validity, of the test. Vice-versa, using a “superset” that includes outliers invalidates the test. However, a small percentage of outliers can be managed by considering a robust (Huber) norm \( \|f - W \rho\|_H \) instead of the \( \mathbb{L}^2 \) norm. Therefore, one could consider the sequential hypothesis testing problem, starting from each \( x_i \in [a = b] \) as an hypothesis, then “growing” the region by one sample, and repeating the test. Note that the optimization has to be solved at each step.\(^{13}\) As a first-order approximation, one can fix the interval \([a, b]\) and accept a less powerful test (if that is a subset of the actual domain) or a test corrupted by outliers (if it is a superset). This is, in fact, done in all local feature-based registration or correspondence methods, and even in region-based segmentation of textures, where statistics must be pooled in a region.

While (d) is largely an open question, (e) follows directly from classical sampling considerations, as described in Sect. A.1.

### B.3 Anti-aliasing descriptors

In the case of matching images under nuisance variability, it has been shown [13] that the ideal descriptor computed at a location \( x \) is not a vector, but a function that approximates the likelihood, where the nuisances are marginalized. In practice the descriptor is approximated with a regularized histogram, similar to SIFT (1). In this case, anti-aliasing corresponds to a weighted average across different locations, scales and domain sizes. But the averaging in this case is simply accomplished by pooling the histogram across different locations and domain-sizes, as in (2). The weight function can be design to optimize the tradeoff between sensitivity and discrimination, although in Sect. 4 we use a simple uniform weight.

\(^{12}\) Purported superiority of “dense SIFT” (regularly sampled at thousands of location) compared to ordinary SIFT (at tens or hundreds of detected location), as reported in few empirical studies, is misleading as comparison has to be performed for a comparable number of samples.

\(^{13}\) In this interpretation, the test can be thought of as a setpoint change detection problem. Another interpretation is that of (binary) region-based segmentation, where one wishes to classify the range of a function \( f - W \rho \) into two classes, with values coming from either \( \rho \) or the background, but the thresholds is placed on the domain of the function \([a, b] \). Of course, the statistics used for the classification depend on \( a, b \) so this has to be solved as an alternating minimization, but it is a convex one [7].
To see how pooling can be interpreted as a form of generalized anti-aliasing, consider the function $f$ sampled on a discretized domain $f(x_i)$ and a neighborhood $B_\sigma(x_i)$ (for instance the sampling interval). The pooled histogram is

$$p_{x_i}(y) = \frac{1}{\sigma} \sum_{x_j \in B_\sigma(x_i)} \delta(y - f(x_j))$$

(9)

whereas the anti-aliased signal (for instance with respect to the pillbox kernel) is

$$\phi(x_i) = \frac{1}{\sigma} \sum_{x_j \in B_\sigma(x_i)} f(x_j)$$

(10)

The latter can be obtained as the mean of the former

$$\phi(x_i) = \sum_y yp_{x_i}(y)$$

(11)

although former can be used for purposes other than computing the mean (which is the best estimate under Gaussian ($\ell^2$) uncertainty), for instance to compute the median (corresponding to the best estimate under uncertainty measured by the $\ell^1$ norm), or the mode:

$$\hat{f}(x_i) = \arg \max_y p_{x_i}(y).$$

(12)

The approximation is accurate only to the extent in which the underlying distribution $p_x(y) = p(f(x) = y)$ is stationary and ergodic (so the spatially pooled histogram approaches the density), but otherwise it is still a generalization of the weighted average or mean.

This derivation also points the way to how a descriptor can be used to synthesize images: Simply by sampling the descriptor, thought of as a density for a given class [13, 46]. It also suggests how descriptors can be compared: Rather than computing descriptors in both training and test images, a test datum can just be fed to the descriptor, to yield the likelihood of a given model class [16], without computing the descriptor in the test image.

C Extensions

Domain-scale pooling can be applied beyond SIFT. Indeed, straightforward extensions of other descriptors can be obtained simply by computing (2) in locations other than a regular $4 \times 4$ grid. For instance, DSP-HOG can be obtained in a straightforward manner this way. Similarly, when the cells are not computed on a lattice, but on a tree, we can obtain DSP-DPM. Also, descriptors other than SIFT, but that still perform spatial pooling, can easily be extended to their DSP-version.

The analysis reported in this appendix can also be extended to two-dimensional signals, albeit at the cost of increased notational complexity.

Also, the pooling approach proposed pertains to the (local) marginalization of group transformations. While this is done routinely for the translation group, other groups have been neglected. We have shown how to perform the construction for the scale (semi-)group, but the same can be done for planar translations, affine deformations, projective, all the way to planar diffeomorphisms [41] or epipolar transformations induced by rigid motion. Of course, the larger the group the more difficult it is to sample from it. However, actual samples in the form of multiple views of the same scene, as available from a (ideally sufficiently exciting) training set, or synthesized from a partial reconstruction, have been championed by [13].

In particular, thus far we have assumed that viewpoint changes were restricted to the translation group. Rotations in space can be decomposed into a component parallel to the optical axis (in-plane rotation) and one orthogonal to it (out-of-plane rotation) with two degrees of freedom. Such “out-of-plane” rotations can be sampled within the visibility cone, and the same procedure described here applied. This is one of our directions of future work.

Additional future work concerns the optimization of sampling and pooling intervals, the choice of sampling (detector) and pooling (descriptor) in addition to the pooling kernel.