Self-Similar Sketch

Andrea Vedaldi and Andrew Zisserman

Department of Engineering Science, University of Oxford
{vedaldi,az}@robots.ox.ac.uk

Abstract. We introduce the self-similar sketch, a new method for the extraction of intermediate image features that combines three principles: detection of self-similarity structures, nonaccidental alignment, and instance-specific modelling. The method searches for self-similar image structures that form nonaccidental patterns, for example collinear arrangements. We demonstrate a simple implementation of this idea where self-similar structures are found by looking for SIFT descriptors that map to the same visual words in image-specific vocabularies. This results in a visual word map which is searched for elongated connected components. Finally, segments are fitted to these connected components, extracting linear image structures beyond the ones that can be captured by conventional edge detectors, as the latter implicitly assume a specific appearance for the edges (steps). The resulting collection of segments constitutes a "sketch" of the image. This is applied to the task of estimating vanishing points, horizon, and zenith in standard benchmark data, obtaining state-of-the-art results. We also propose a new vanishing point estimation algorithm based on recently introduced techniques for the continuous-discrete optimisation of energies arising from model selection priors.

Key words: self-similarity, feature detector, vanishing point estimation, UFL

1 Introduction

Almost all computer vision methods start by computing features of the image. This is done in addressing geometric tasks such as three dimensional reconstruction from multiple views, as well as semantic tasks such as the recognition of natural object categories. Useful features extract stable image structures which are relevant to the task at hand, factoring the useful information from nuisances of the imaging process. A typical example are the many co-variant image region detectors [1–3] and the corresponding invariant descriptors [4] used in wide-baseline matching to identify fragments of 3D dimensional scenes regardless of viewpoint.

Due to the lack of a theory that can indicate what an optimal feature design should be, one usually looks for reasonable properties such as viewpoint invariance [5], robustness, and speed. Standard feature *detectors* such as Harris' corners [6], Canny's edges [7], the Laplacian and Hessian detectors [8], and

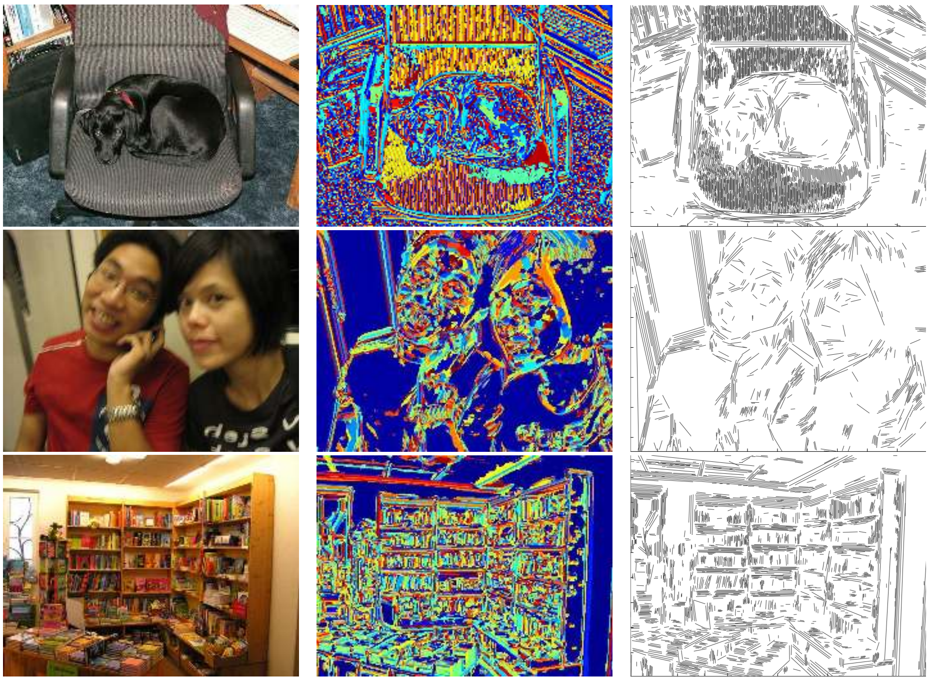


Fig. 1. Self-similar sketches. The figure shows a few example images, the quantised local descriptors, and the fitted line segments constituting the sketch. The calculation is straightforward and relatively efficient.

their affine co-variant extensions [1, 2], look for simple structures such as corners, blobs, and steps. In these cases, the appearance of the detected structures is defined analytically by looking at the extrema of one or more operators of the image scale space. Alternatively, it can be *learned* from example data [9, 10] to go past the restrictions of the analytical approach.

In this work we propose a novel design for feature detectors which does *not* require defining (either analytically or through example data) the *appearance* of the detected structures at all. Our approach, which we call *self-similar sketch*, combines in a novel way three known principles: (i) self-similarity [11], (ii) nonaccidental alignment [12–14], and (iii) instance-specific modelling [15]. The idea of self-similarity, as proposed by [11], is to abstract from the image appearance by searching for occurrences of visually similar structures within an image. Spatial arrangements of self-similar structures are then recorded and used as feature *descriptors*. By contrast, the self-similar sketch looks for nonaccidental arrangements (in the following examples straight line segments) of self-similar structures and use these as a basis for feature *detectors*. Moreover, the method used to detect the self-similar structures, *i.e.,* computing an image-specific visual vocabulary [16] (Sect. 2), can be seen as estimating an appearance model of each detected structure on an instance-by-instance basis.

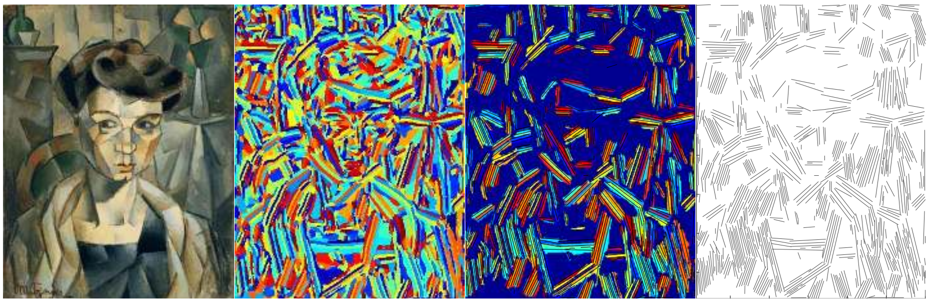


Fig. 2. Construction of the self-simlar sketch. From left to right: input image, label $l(u,v)$ associated to each pixel (different colours correspond to different labels), regions selected as elongated overlaid by the corresponding line segment, and the final sketch.

The paper explores a particularly simple example of the self-similar sketch construction that can be summarised in a few words: (i) local descriptors are extracted densely from the image, (ii) these are discretised based on an image-specific vocabulary, and (iii) the aligned occurrences of identical visual words are extracted as line segments. The result can be thought of as a sketch of the image abstracting away from the details of the local appearance while capturing shape by coarse straight strokes (Fig. 1, 2, Sect. 2).

Compared to traditional feature designs, perhaps the most striking difference is that the self-similar sketch does not attempt to characterise the *appearance* of the detected structure (*e.g.,* as blobs or steps). Instead, the feature appearance is completely instance-dependent, and the features are defined solely based on a nonaccidental alignment principle. The details of the construction are reported in Sect. 2.

As an example application, Sect. 4 demonstrates using the self-similar sketch for the extraction of vanishing points and the estimation of the horizon location on standard benchmark data. For vanishing point estimation, a method inspired by [17, 18] (Sect. 3) is used to simultaneously group of line segments into vanishing points, determine the number and locations of such points, and reject potential line outliers. By carefully modelling the uncertainty of line detector output (Sect 3.3), this method is competitive with state-of-the-art algorithms even when standard edge detectors are used (Sect. 4). When the algorithm is used in combination with the self-similar sketch, however, the performance is improved significantly and results may exceed the state-of-the-art by a margin.

2 Self-similar sketch

This section describes in detail the extraction of the self-similar sketch outlined in Sect. 1. Fig. 2 illustrates the main steps.

Given an input image $I(u,v)$, $(u,v) \in \Omega$, where Ω is a discrete set of pixel locations, the construction starts by extracting a dense set of local appearance descriptors $\{\Psi(u,v), (u,v) \in \Omega\}$ from the image. While many different choices of $\Psi(u,v)$ are possible, in this paper we consider SIFT features [4] as these can be extracted very efficiently at a dense set of locations if their orientation and scale are fixed. Very low-contrast SIFT descriptors are mapped to the null vector to avoid wasting resources by coding unstable image structures in the next step.

The descriptors $\{\Psi(u,v) : (u,v) \in \Omega\}$ are then quantised in a small image-specific vocabulary $\{\mathbf{c}_1, \dots, \mathbf{c}_k\}$ of $k \approx 100$ visual words by using k-means clustering. While the number of descriptors is potentially large, the small size of the vocabulary means that even a simple implementation of the k-means algorithm is usually sufficient to complete the computation quickly (it is also possible to subsample the set of descriptors used to estimate the k-means centres).

After quantisation, line segments are efficiently extracted by using a method analogous to [19]. First, each image pixel (u,v) is assigned the label $l(u,v)$ corresponding to the visual word $\mathbf{c}_{l(u,v)}$ closest to the local descriptor $\Psi(u,v)$. Then the connected components of the label map $l(u,v)$ are extracted, in our example using the standard 4-neighbours topology. The area $A(R)$ and major $\mu_1(R)$ and minor $\mu_2(R)$ axis of inertia of each connected component $R \subset \Omega$ are computed (this can be done efficiently in a single pass over the image by using appropriate accumulators), and the regions R that have sufficiently large area $A(R) \geq A_0$ and aspect ratio $\mu_1(R)/\mu_2(R) > r$ are deemed as *elongated* and marked as detected.

Once these elongated regions are extracted, a line segment is fitted to each of them. The segment passes through the centroid of the region R, is aligned to the major axis of inertia of the region, and extends to touch a bounding box tightly containing the region itself. In the following, each segment will be represented by the two extrema $(\mathbf{x}_1, \mathbf{x}_2)$. The collection of all the segments extracted in this manner constitutes the self-similar sketch. Example results are reported in Fig. 2 and Sect. 4.

3 Vanishing points from the self-similar sketch

This section discusses an example application of the self-similar sketch, namely the automatic extraction of vanishing points and related geometric entities such as the horizon. Recall that a *vanishing point* is an image point where the projection of parallel three dimensional lines converge. The estimation of the image vanishing points [20–26] is a standard step in several geometry-based applications. For example, in a so-called Manhattan world one has ideally three groups of three dimensional parallel lines, corresponding to the vertical direction (the side of buildings) and two orthogonal ones, parallel to the ground (the bases of buildings), forming an orthonormal system. Thus, by identifying these characteristic directions, it is possible to estimate the camera orientation. In a slightly more complex scenario, there are a number of directions that are parallel to the ground but not necessarily orthogonal, corresponding to vertical structures whose base may not be aligned. In this case, vanishing points can still be useful to estimate other important geometric parameters, such as the location of the horizon. In particular, the estimation of the horizon has been shown to be a useful cue for higher level tasks, including object categorisation and scene interpretation [27].

3.1 Geometry

This section reviews briefly the geometry of vanishing points and the relative equations. Given a point $(u, v) \in \Omega$ in the image reference frame, its coordinates $\mathbf{x} \in \mathbb{R}^3$ in the camera reference frame are given by

$$\mathbf{x} = \begin{bmatrix} \rho(u - c_u) \\ \rho(v - c_v) \\ 1 \end{bmatrix}$$

where ρ is a pixel size in units of focal length and (u_c, v_c) are the coordinates of the camera principal point (in pixels). When the pixel size and the principal points are unknown, a sensible choice [21] is to assume that the principal point lies at the centre of the image and that $\rho = 2/W$, where W is the image width in pixels (this corresponds to assuming a field of view is of 90 degrees).

A line segment is defined by two extrema $(\mathbf{x}_1, \mathbf{x}_2)$ in the image plane. The corresponding line can be obtained by intersecting the image plane with the plane π passing through $\mathbf{x}_1, \mathbf{x}_2$ and the camera centre $\mathbf{0}$. Therefore, the line can be represented as the unit vector $\ell \in \mathbb{S}^2$ normal to the plane π, where $\mathbb{S}^2$ denotes the Gaussian (unit) sphere. This vector is given by

$$\ell = \frac{\hat{\mathbf{x}}_1 \mathbf{x}_2}{\|\hat{\mathbf{x}}_1 \mathbf{x}_2\|} \tag{1}$$

where $\hat{\cdot}$ denotes the hat operator (*i.e.,* $\hat{\mathbf{x}}_1 \mathbf{x}_2 = \mathbf{x}_1 \times \mathbf{x}_2$).

Let $\ell_1, \ldots, \ell_n$ be lines whose image converges to the same vanishing point. Then the vectors $\ell_1, \ldots, \ell_n$ belong to a plane π' passing through the camera centre. If $\mathbf{v} \in \mathbb{S}^2$ is the vector normal to π', then all the vectors ℓ_i are orthogonal to it, *i.e.,* $\langle \ell_i, \mathbf{v} \rangle = 0$. Moreover, the image of the vector $\mathbf{v}$ is the vanishing point.

Given n lines $\ell_1, \ldots, \ell_n$, the goal is to associate them to m vanishing points $\mathbf{v}_i, \ldots \mathbf{v}_m$ in such a way that the equations

$$\langle \mathbf{v}_{q_i}, \ell \rangle = 0, \quad i = 1, \ldots, n \tag{2}$$

are satisfied. Here $q_i \in \{1, \ldots, m\}$ are n label assignments mapping the n lines to up to m vanishing points. In general, not only the associations q_i and vanishing points $\mathbf{v}_i$ are unknown, but also the number m of the latter must be determined. Moreover, measurements are affected by noise and can be contaminated by outliers. The next section introduces an effective statistical model that can handle all this automatically.

3.2 Statistics and objective function

This section introduces a statistical model for associating lines to vanishing points, estimating the number and location of these, and handling potential outliers in the measurements. The association is represented by n label assignments

$q_i \in \{1, \dots, m, m+1\}$, where the additional label $m+1$ indicates that a line is an outlier (by convention we set the corresponding vanishing point vector $\mathbf{v}_{m+1} = 0$ to zero). The goal is to find a small number of vanishing points that can explain all the lines that are inliers. This can be obtained by trading off the number of vanishing point used to "explain" the lines and the likelihood of the fit, as expressed by the minimisation of the energy function

$$E(q_1, \dots, q_n; \mathbf{v}_1, \dots, \mathbf{v}_m) = -\sum_{i=1}^{n} \log p(\ell_i | \mathbf{v}_{q_i}, \sigma_i) + \gamma \sum_{j=1}^{m+1} [\exists i : q_i = j]. \tag{3}$$

The first term is the negative log-likelihood of the vanishing points given the measured lines and the second term is the number of vanishing points that are associated to at least one line. Choosing a large parameter γ discourages selecting too many vanishing points, avoiding overfitting the data. This formulation is a special case of, for example, [17].

The probability density $p(\ell|\mathbf{v}, \sigma)$ is the likelihood of the vanishing point $\mathbf{v}$ given the measured line ℓ and accounts for the measurement error, including the outlier case $\mathbf{v} = 0$. According to (2), in the ideal case all the line vectors ℓ_i are orthogonal to $\mathbf{v}_{q_i}$. A simple model [21] is therefore to assume that $\langle \ell_i, \mathbf{v} \rangle$ has a null mean Gaussian distribution, *i.e.,*

$$-\log p(\ell|\mathbf{v}, \sigma) = \begin{cases} \frac{1}{2}\langle \ell, \mathbf{v} \rangle^2/\sigma^2 + \frac{1}{2}\log 2\pi\sigma, & \mathbf{v} \neq 0, \\ \frac{1}{2}\log 2\pi\sigma_0, & \mathbf{v} = 0 \quad \text{(outlier)}. \end{cases} \tag{4}$$

Here the standard deviation σ should be proportional to the uncertainty induced by measurement errors. The parameter σ_0 is a nominal standard deviation corresponding to the outlier case, expressed in this way to make tuning more intuitive. The next section introduces a simple yet effective method for tuning σ based on the uncertainty of each line segment.

3.3 Measurement model

This section suggests how to choose the standard deviations σ_i in (3) in order to reflect the accuracy of the line segment detection. When the line segments are obtained as discussed in Sect. 2, they are affected by an error $\tilde{\mathbf{x}}_i = \mathbf{x}_i + \mathbf{e}_i$, where $\mathbf{e}_i = (e_{1i}, e_{2i}, 0)^\top$ is a small displacement in the image plane. The modulus of the error can be bounded by a parameter $\delta \geq \|\mathbf{e}_i\|$ equal to half the width of the elongated region R used to extract that segment, which for simplicity is assumed to be proportional to its minor axis of inertia $\mu_2(R)$ (Sect. 2). Let $\tilde{\ell} = \hat{\tilde{\mathbf{x}}}_1\mathbf{x}_2/\|\hat{\tilde{\mathbf{x}}}_1\mathbf{x}_2\|$ be the line perturbed by the error at the extrema of the segment. The uncertainty on the projection of the line ℓ on the vanishing point vector $\mathbf{v}$ is then bounded by the Cauchy-Schwartz inequality:

$$|\langle \mathbf{v}, \tilde{\ell} \rangle - \langle \mathbf{v}, \ell \rangle| \leq \|\mathbf{v}\| \|\tilde{\ell} - \ell\| = \|\tilde{\ell} - \ell\|.$$

The difference between the perturbed and exact line is given by

$$\tilde{\ell} - \ell = \frac{\hat{\tilde{\mathbf{x}}}_1\tilde{\mathbf{x}}_2}{\|\hat{\tilde{\mathbf{x}}}_1\tilde{\mathbf{x}}_2\|} - \frac{\hat{\mathbf{x}}_1\mathbf{x}_2}{\|\hat{\mathbf{x}}_1\mathbf{x}_2\|} \approx \frac{\hat{\tilde{\mathbf{x}}}_1\tilde{\mathbf{x}}_2 - \hat{\mathbf{x}}_1\mathbf{x}_2}{\|\hat{\mathbf{x}}_1\mathbf{x}_2\|} = \frac{\hat{\mathbf{x}}_1\mathbf{e}_2 - \hat{\mathbf{x}}_2\mathbf{e}_1 + \hat{\mathbf{e}}_2\mathbf{e}_1}{\|\hat{\mathbf{x}}_1\mathbf{x}_2\|}.$$

Since $\hat{\mathbf{x}}_1 \mathbf{e}_2 = (-e_{22}, e_{12}, x_{11}e_{22} - x_{21}e_{12})^\top$, then, in the worst case, one obtains that $e_{12} = \delta x_{21}/\sqrt{x_{11}^2 + x_{21}^2}$ and $e_{21} = -\delta x_{11}/\sqrt{x_{11}^2 + x_{21}^2}$, so that $\|\hat{\mathbf{x}}_1 \mathbf{e}_2\|^2 < \delta^2(1 + x_{11}^2 + x_{21}^2)$. Since $\|\hat{\mathbf{e}}_2 \mathbf{e}_1\|^2 \leq \delta^4$, one obtains

$$|\langle \mathbf{v}, \tilde{\ell} \rangle - \langle \mathbf{v}, \ell \rangle| \leq \|\tilde{\ell} - \ell\| \leq \delta \frac{\sqrt{\|\mathbf{x}_1\|^2 + \|\mathbf{x}_2\|^2 + \delta^2}}{\|\mathbf{x}_1 \times \mathbf{x}_2\|}. \tag{5}$$

It is then natural to set the standard deviation σ in the likelihood (4) proportional to the right-hand side of (5).

3.4 Optimisation

The optimisation of the energy function (3) is quite challenging as one has to simultaneously allocate vanishing points to lines as well as determine how many should be used. This is the same problem addressed, for example, by the PEARL algorithm of [17, 18], and in fact a special case which reduces to the *uncapacitated facility location* (UFL [18]) algorithm as there are no pairwise terms in the energy. While not explored here, pairwise terms could be used in the estimation of vanishing points in order to encourage image lines that are nearly parallel and spatially close to converge to the same vanishing point. The algorithm combines four steps, initialisation, α-expansion, re-estimation, and re-sampling, as detailed next.

Initialisation. The algorithm starts by considering a large set of candidate vanishing points $\mathbf{v}_1, \dots, \mathbf{v}_m$. These can be sampled from data in various ways. In our implementation, one vanishing point is generated from each line ℓ_i by setting $\mathbf{v}_i \propto (0, 0, 1) \times \ell_i$. This choice corresponds to initialising a vanishing point at infinity in the direction of each line ℓ_i.

α-expansion. Given a label $\alpha \in \{1, \dots, m, m+1\}$ to expand, one searches which label assignments q_i should switch to α in order to maximally decrease the energy (3). There are two cases to be considered:

- If the label α is already active (*i.e.,* $\exists i : q_i = \alpha$) switching any assignment q_i to α does not pay the cost γ of activating a new label. Given this observation, there are two cases for which switching improves the likelihood. The first one is that the likelihood of α for the line ℓ_i is better:

$$-\log p(\ell_i | \mathbf{v}_\alpha, \sigma_i) < -\log p(\ell_i | \mathbf{v}_{q_i}, \sigma_i). \tag{6}$$

  The second one is that switching *all the current assignments* $I = \{i : q_j = \beta\}$ to some label $\beta \neq \alpha$ back to α lowers the energy accounting for the additional reward γ obtained by making the label β inactive:

$$-\sum_{i \in I} \log p(\ell_i | \mathbf{v}_\alpha, \sigma_j) < -\sum_{i \in I} \log p(\ell_i | \mathbf{v}_{q_i}, \sigma_i) + \gamma, \tag{7}$$

- If on the contrarily α is inactive in the current labelling (*i.e.,* $\forall i : q_i \neq \alpha$), then conditions similar to (6) and (7) still apply, but one needs to check whether the overall improvement is larger than the cost γ of activating the new label α before switching.

Once a new assignment q^α is obtained from q by using these rules, the move is accepted if it lowers the energy:

$$E(q_1^\alpha, \dots, q_n^\alpha; \mathbf{v}_1, \dots, \mathbf{v}_m) < E(q_1, \dots, q_n; \mathbf{v}_1, \dots, \mathbf{v}_m).$$

Re-estimation. After labels have been reassigned by an α-expansion step, one can further improve the energy by updating the vanishing points to optimally match the lines currently assigned to them. Given the likelihood (4), this amounts to calculating

$$\mathbf{v}_j^* = \operatorname*{argmin}_{\mathbf{v} \in \mathbb{S}^2} \sum_{i : q_i = j} \frac{\langle \ell_i, \mathbf{v} \rangle^2}{\sigma_i^2} \tag{8}$$

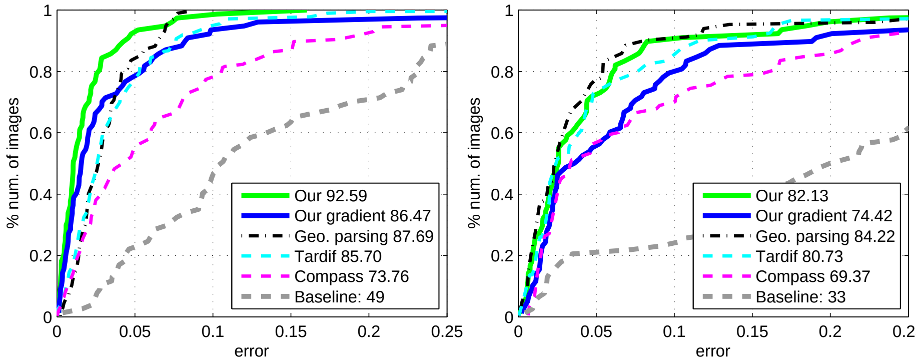
Then the vanishing point $\mathbf{v}_j^*$ is obtained [21] as the unit eigenvector of $M^\top M$, where the columns of the matrix M are the vectors ℓ_i/σ_i, $i : q_i = j$.

Re-sampling. After a few iterations of α-expansion and re-estimation, the algorithm may converge to a locally optimal solution. This can be further improved by proposing new candidate vanishing points. In practice, the next time an inactive label α is expanded, rather than using the current value of $\mathbf{v}^\alpha$, this can be replaced by the vanishing point obtained by intersecting two lines ℓ_i, ℓ_j selected at random, or by intersecting the lines $\{\ell_i : q_i = j \vee q_i = k\}$ obtained by merging two active labels j and k (in this case $\mathbf{v}^\alpha$ is computed using (8)).

4 Experiments

This section evaluates empirically the self-similar sketch applied to the task of estimating vanishing points. The comparison includes: (i) the algorithm of Sect. 3 with the self-similar sketch (Sect. 2), (ii) the algorithm of Sect. 3 with lines obtained as in Sect. 2 but based on the image gradient (which reduces to [19] and is used by Video Compass [21]), (iii) the current state-of-the-art geometric parsing algorithm of [24, 28] based on Canny's edges, (iv) Tardif's method [23], and (v) Video Compass [21].

Video Compass [21] is a carefully tuned expectation-maximisation (EM) algorithm that, similarly to the method of Sect. 3.4, fits lines to vanishing points on the Gauss sphere, hence using an algebraic error. *Tardif's method* [23] uses J-Linkage to non-iteratively associate line segments to vanishing points and refines the solution by using EM iterations, measuring errors directly on the image plane. *Geometric parsing* [24] proposes and integrated energy formulation that simultaneously groups edges into lines, lines into vanishing points, and use the latter to estimate the horizon an zenith.



(a) (b)

Fig. 3. Quantitative evaluation of the horizon estimation. (a) York Urban and (b) Eurasian Cities datasets. The figure compares our estimation algorithm combined with the self-similar sketch, the same with gradient-based features, three competing methods, and the baseline obtained by setting the horizon to be an horizontal line in the image centre. See the text for details.

Datasets. The methods are evaluated on the same datasets and with the same protocol of [24], and in fact we report their results for all but the proposed methods. The methods are evaluated on two datasets: (i) York Urban [22] consisting of 102 images mostly following the Manhattan world assumption (three main orthogonal vanishing points) taken indoor and outdoor around the same location and (ii) the significantly harder Eurasian dataset of [24], including scenes from different parts of the world (hence with different appearance statistics), more varied viewpoints, and poorer fit to the Manhattan assumption. Both datasets come with a few accurately hand annotated systems of vanishing lines for each image that are used to estimate the ground truth parameters during evaluation. Images are split into validation and testing as explained in [24].

Task and performance metric. [24] proposes as performance metric the quality of the recovered horizon line, as the latter can be obtained directly from un-calibrated images (camera calibration is not available for Eurasian Cities). Following the protocol in [24], the horizon is estimated into two steps: first the zenith (vertical direction) is used to estimate the orientation of the horizon (as the zenith is orthogonal to it) and then the other vanishing points are used to estimate the horizon offset. In principle, in estimating the horizon offset only the vanishing points corresponding to three dimensional lines parallel to the ground should be used. While the sophisticated model in [24] explicitly identifies such vanishing points, here we use the simple heuristic of letting all the vanishing points vote for the horizon based on their mass (number of line segments converging to them). Performance is reported in term of the percentage of test images that achieve an error smaller or equal than a given threshold, obtaining corresponding performance curves as the threshold is varied (Fig. 3). We also

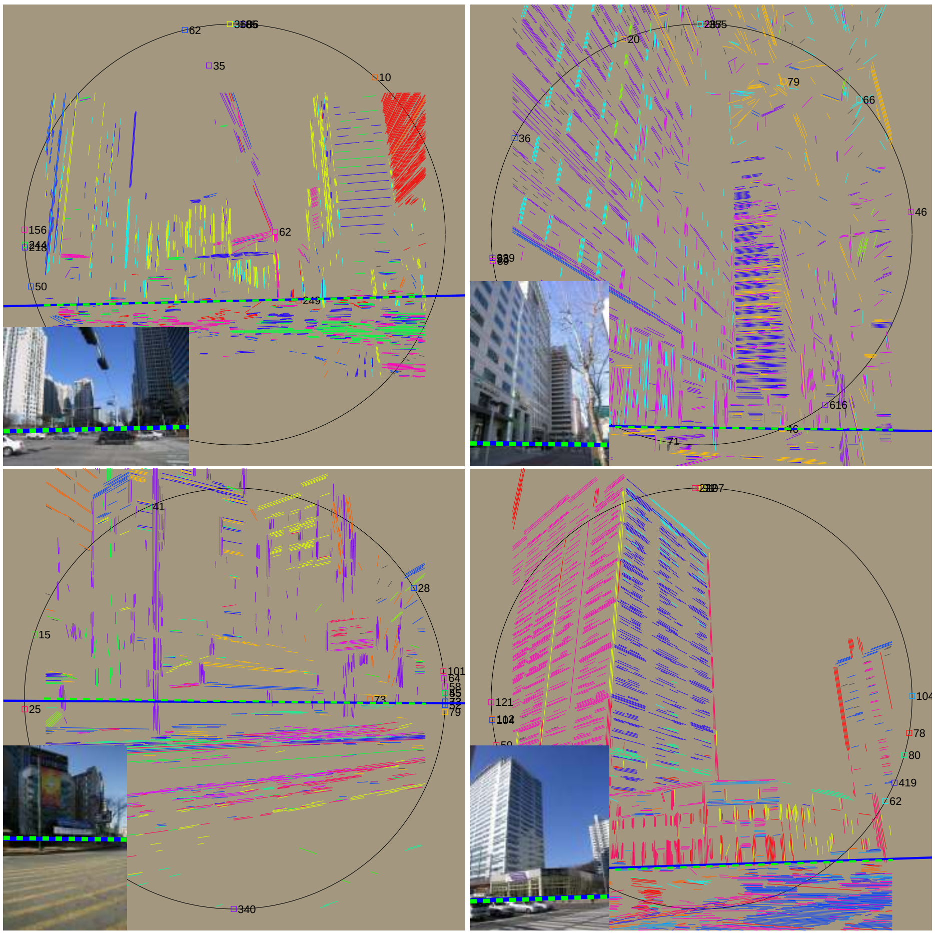


Fig. 4. Vanishing points and horizon estimation. Example images, vanishing points, and horizon (in solid blue the estimated one and in dashed green the ground truth one) on the Eurasian Cities dataset. The coloured line segments correspond to the self-similar sketch elements that converge to one of the vanishing points. Each vanishing point is marked by a square along with the number of segments converging to it. For the vanishing points outside the circle only the direction (with respect to the image centre) is shown.

report a numerical value as the percentage of area under the curve in the subset $[0, 0.25] \times [0, 1]$ (*i.e.,* focusing on the low-error region of the plots).

4.1 Implementation details

For the fast extraction of dense SIFT features and k-means clustering we use the public implementation of these two algorithms in VLFeat [29] with default

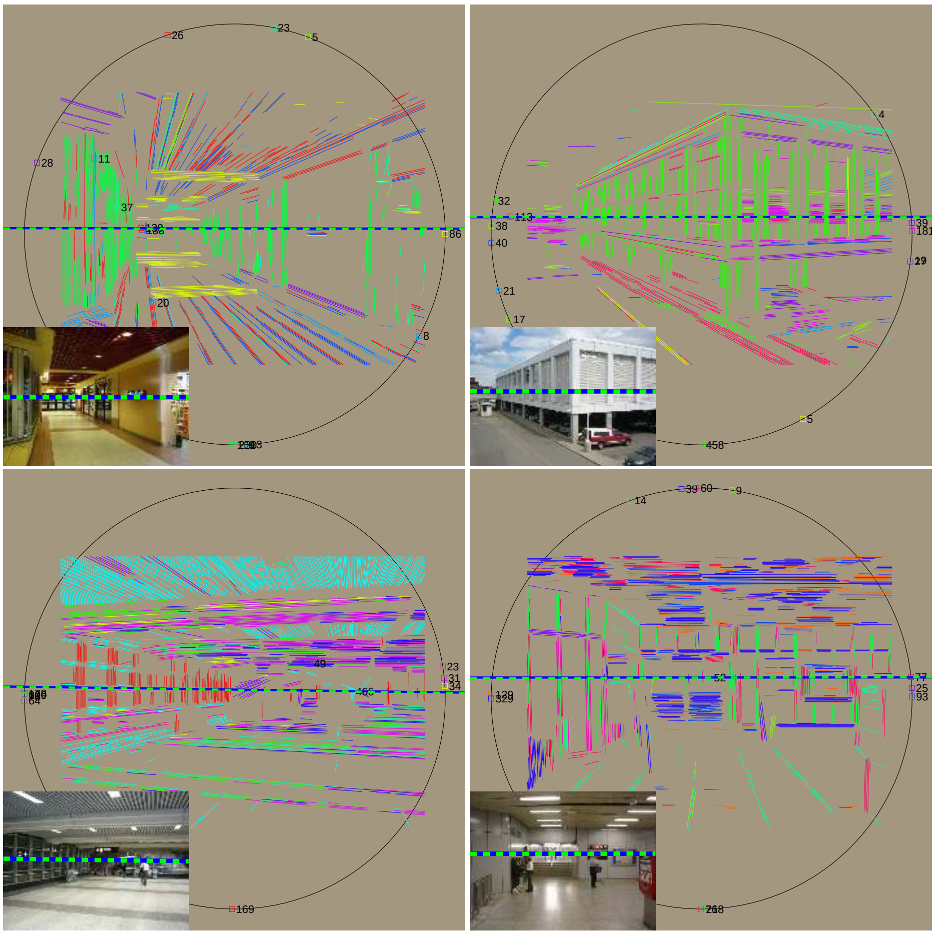


Fig. 5. Vanishing points and horizon estimation. The same of Fig. 4 but for the York Urban dataset.

values. No image smoothing is applied before extraction of the SIFT features and the size of a spatial bin is set to two pixels. The number of visual words for each image-dependent vocabulary was set to 100, although different values (*e.g.*, 50 or 150) did not change results substantially. The minimum area of an elongated connected component was set to 17 and 20 pixels respectively on the Eurasian Cities and the York Urban dataset and the minimum ratio between minor and major moment of inertia of the regions was set to 7 and 9 respectively. Increasing such values may select a more reliable but smaller set of linear structures in the images and can affect performance somewhat due to the limitations of the model of Sect. 3.2 and the fact that the optimisation method of Sect. 3.4 might still be confused by too many outliers and get stuck into a bad local optimum. The computation of the sketch requires only a few seconds on a standard laptop computer with an un-optimised MATLAB implementation.

The self-similar sketch is compared to the method of [19] for the extraction of line segments from an image. [19] can be implemented simply by changing the generation of the label map $l(u, v)$ in Sect. 2 to be the quantised gradient orientation. The best parameters for the extraction of this feature amounted to smooth the images by a Gaussian kernel of variance 0.5 pixels and to quantise the gradient orientation in 12 bins. The optimal parameters for the selection of the elongated connected components were found to be the same as for the self-similar sketch.

4.2 Results

Fig. 4 reports example results on the Eurasian dataset and Fig. 5 on the York Urban dataset. Fig. 3 reports the quantitative comparison of the various methods. The baseline method of Sect. 3 with the gradient-based line segments of [19] is already quite competitive, demonstrating the effectiveness of the model of Sect. 3.2 combined with the optimisation method of Sect. 3.4. The most important result, however, is that switching from the gradient-based edges to the self-similar sketch significantly boosts performance. In particular, our complete method outperforms more specialised techniques such as the geometric parsing of [24] on the York Urban dataset and is very close to it on the Eurasian Cities. The reason for the improved quality of these results is the fact that the self-similar sketch is able to extract many additional linear image structures beyond standard edge detectors. By looking at the gradient, these methods in fact assume implicitly that edges correspond to steps of the intensity profile. By contrast, the self-similar sketch does not make any particular assumption on the local appearance of linear structures.

5 Summary and future work

We have introduced the idea of self-similar sketch, a method for detecting image features without committing to their appearance beforehand. The self-similar sketch looks for self-similar structures in an image that happen to be arranged in a non-accidental manner. A simple implementation of this idea was obtained by clustering SIFT features with an image-dependent vocabulary and looking for collinear occurrences of identical visual words. The resulting linear structures are more reliable than standard gradient-based edge extraction methods when applied to the task of estimating vanishing points, zenith, and horizon in natural images. As part of this evaluation, recent methods for the optimisation of discrete-continuous energies arising from automatic model selection problems have been found to perform very well.

This paper has examined only a simple example of self-similar sketch. Future directions include exploring different ways of computing self similarity and the use of other type of nonaccidental cues. Particularly interesting cases include the extraction of curved structures and of groups of collinear, parallel ones. Most importantly, we plan to explore the use of the sketch in semantic image analysis tasks such as object detection and image categorisation. A simple way to do so is to compute descriptors on top of the detected sketch features. By enabling abstracting from instance-specific details, this would be similar to the original self-similar sketch descriptors [11], while incorporating the notion of alignment/non-accidentally. The qualitative examples of Fig. 1 suggest that this may be a promising direction.

References

1. Baumberg, A.: Reliable feature matching across widely separated views. In: CVPR, IEEE Press, New York (2000) 1774–1781
2. Mikolajczyk, K., Schmid, C.: An affine invariant interest point detector. In Heyden, A., Sparr, G., Nielsen, M., Johansen, P., eds.: ECCV (1). Volume 2350 of LNCS., Springer, Heidelberg (2002) 128–142
3. Matas, J., Chum, O., Urban, M., Pajdla, T.: Robust wide baseline stereo from maximally stable extremal regions. In Rosin, P.L., Marshall, A.D., eds.: BMVC, British Machine Vision Association (2002)
4. Lowe, D.G.: Object recognition from local scale-invariant features. In: ICCV. (1999) 1150–1157
5. Vedaldi, A., Soatto, S.: Features for recognition: Viewpoint invariance for non-planar scenes. In: ICCV, IEEE Press, New York (2005) 1474–1481
6. Harris, C., Stephens, M.: A combined corner and edge detector. In: Proc. of The Fourth Alvey Vision Conference. (1988) 147–151
7. Canny, J.: A computational approach to edge detection. IEEE Trans. on Patt. Analysis and Machine Intell. **8** (1986)
8. Lindeberg, T.: Principles for automatic scale selection. Technical Report ISRN KTH/NA/P 98/14 SE, Royal Institute of Technology (1998)
9. Arbelaez, P., Maire, M., Fowlkes, C., Malik, J.: Contour detection and hierarchical image segmentation. IEEE Trans. on Patt. Analysis and Machine Intell. **33** (2011) 898–916
10. Rosten, E., Drummond, T.: Machine learning for high-speed corner detection. In Leonardis, A., Bischof, H., Pinz, A., eds.: ECCV (1). Volume 3951 of LNCS., Springer, Heidelberg (2006) 430–443
11. Shechtman, E., Irani, M.: Matching local self-similarities across images and videos. In: CVPR, IEEE Press, New York (2007)
12. Biederman, I.: Recognition-by-components: A theory of human image understanding. Psychological Review **94** (1987)
13. Agin, G.J., Binford, T.O.: Computer description of curved objects. IEEE Trans. Comp.s **25** (1976) 439–449
14. Gibson, B.M., nad F. Gosselin, O.F.L., Schyns, P.G., Wasserman, E.A.: Nonaccidental properties underlie shape recognition in mammalian and nonmammalian vision. Current Biology **17** (2007)
15. Jojic, N., Caspi, Y.: Capturing image structure with probabilistic index maps. In: CVPR (1). (2004) 212–219
16. Deselaers, T., Ferrari, V.: Global and efficient self-similarity for object classification and detection. In: CVPR, IEEE Press, New York (2010) 1633–1640

17. Isack, H.N., Boykov, Y.: Energy-based geometric multi-model fitting. Int. Journal of Comp. Vision **97** (2012) 123–147
18. Delong, A., Osokin, A., Isack, H.N., Boykov, Y.: Fast approximate energy minimization with label costs. Int. Journal of Comp. Vision **96** (2012) 1–27
19. Kahn, P., Kitchen, L.J., Riseman, E.M.: A fast line finder for vision-guided robot navigation. IEEE Trans. Patt. Anal. Mach. Intell. **12** (1990) 1098–1102
20. Schaffalitzky, F., Zisserman, A.: Planar grouping for automatic detection of vanishing lines and points. Image Vision Comput. **18** (2000) 647–658
21. Kosecká, J., Zhang, W.: Video compass. In Heyden, A., Sparr, G., Nielsen, M., Johansen, P., eds.: ECCV (4). Volume 2353 of LNCS., Springer, Heidelberg (2002) 476–490
22. Denis, P., Elder, J.H., Estrada, F.J.: Efficient edge-based methods for estimating manhattan frames in urban imagery. In Forsyth, D.A., Torr, P.H.S., Zisserman, A., eds.: ECCV (2). Volume 5303 of LNCS., Springer, Heidelberg (2008) 197–210
23. Tardif, J.P.: Non-iterative approach for fast and accurate vanishing point detection. In: ICCV, IEEE Press, New York (2009) 1250–1257
24. Barinova, O., Lempitsky, V.S., Tretiak, E., Kohli, P.: Geometric image parsing in man-made environments. In Daniilidis, K., Maragos, P., Paragios, N., eds.: ECCV (2). Volume 6312 of LNCS., Springer, Heidelberg (2010) 57–70
25. Flint, A., Mei, C., Reid, I.D., Murray, D.W.: Growing semantically meaningful models for visual slam. In: CVPR, IEEE Press, New York (2010) 467–474
26. Flint, A., Murray, D.W., Reid, I.: Manhattan scene understanding using monocular, stereo, and 3d features. In Metaxas, D.N., Quan, L., Sanfeliu, A., Gool, L.J.V., eds.: ICCV, IEEE Press, New York (2011) 2228–2235
27. Hoiem, D., Efros, A.A., Hebert, M.: Closing the loop in scene interpretation. In: CVPR, IEEE Press, New York (2008)
28. Tretiak, E., Barinova, O., Kohli, P., Lempitsky, V.S.: Geometric image parsing in man-made environments. Int. Journal of Comp. Vision **97** (2012) 305–321
29. Vedaldi, A., Fulkerson, B.: VLFeat: an open and portable library of computer vision algorithms. In Bimbo, A.D., Chang, S.F., Smeulders, A.W.M., eds.: ACM Multimedia, ACM (2010) 1469–1472