

Versatile Spectral Methods for Point Set Matching

Alberto Silletti^a, Alessandro Abate^{b,c}, Jeffrey D. Axelrod^c, Claire J. Tomlin^d

^a*Department of Information Engineering, University of Padova,
Padova, Italy*

^b*Delft Center for Systems and Control, TU Delft - Delft University of Technology,
Delft, The Netherlands*

^c*Department of Pathology, Stanford University School of Medicine,
Stanford, CA, USA*

^d*Department of Electrical Engineering and Computer Sciences,
University of California at Berkeley,
Berkeley, CA, USA*

Abstract

This work is concerned with the problem of point set matching over features extracted from images. A novel approach to the problem is proposed, which leverages different techniques from the literature. The approach combines a number of similarity metrics that quantify measures of correspondence between the two sets of features and introduces a non-iterative algorithm for feature matching based on spectral methods. The flexibility of the technique allows its straightforward application in a number of diverse scenarios, thus overcoming domain-specific limitations of known techniques in the literature. The proposed approach is tested in a number of heterogeneous case studies: of synthetic nature; drawn from experimental biological data; and taken from known benchmarks in computer vision.

Key words: Point Set Matching, Transformations and Correspondences, Registration, Similarity Metrics, Spectral Methods, Kernel Methods

1. Introduction

The general problem of point set matching is a fundamental topic in computer vision and is key for the task of registration of multiple images. The problem is defined over pairs of feature sets extracted from images and can be decomposed, as suggested in [5], into two related sub-problems: that of *transformation* between the images, and that of *correspondence* between the features of the two images.

The first problem (transformation) is concerned with finding the mathematical relationship underlying the overall morphing between two successive frames, that is a mapping describing the transformation of the first image into the second frame [19]. This objective is relatively easily obtained when the underlying

Email addresses: alberto.silletti@unipd.it (Alberto Silletti), a.abate@tudelft.nl (Alessandro Abate), jaxelrod@stanford.edu (Jeffrey D. Axelrod), tomlin@eecs.berkeley.edu (Claire J. Tomlin)

23 transformation is rigid: our approach instead does not assume that the transformation between the images
24 is rigid.

25 The second issue (correspondence), which is the focus of the present contribution, deals with the task of
26 finding a quantitative matching between features associated with the two images, which are not necessarily
27 related by a rigid transformation. In the literature this problem is studied either by working directly on
28 the intensity maps associated with the two images [2, 19], or by extracting correspondences between sets of
29 features obtained from the images [5, 13, 14]. It is often the case that the two sets of features are made up
30 of heterogeneously distributed points that are extracted from the original frames.

31 The contribution in [7] employs a brute-force approach to the correspondence problem by exploiting the
32 notion of Hausdorff distance. However the computational burden of the approach arising from combinatorial
33 explosion can be significant. A popular, non-iterative technique uses spectral methods over the abstract
34 structures composed of the feature points – seminal contributions in this area are those in [13, 14], which
35 essentially differ on one major aspect. The former sets up a similarity metric based on *inter*-relationships
36 between pairs of points taken from the two sets, and studies the spectral properties of a matrix that
37 incorporates such a metric. The latter instead compares the spectral properties of two matrices, each
38 of which is intended to describe the shape of the single image and its local relations (*intra*-metrics within
39 the image). The technique is related to other approaches based on matrix spectral analysis for abstract
40 weighted graph matching problems. Similar to the cited work in [13, 14] and to [20], the presented technique
41 is analytic. Alternatively, abstract graph matching can be studied with iterative schemes, such as that in
42 [3], which offsets the computation time related to the scheme with a linear dependence on the graph size.

43 Both [13, 14] have known pitfalls and shortcomings. [13] presents limitations when the rotation or
44 the scaling between the two images is large. To mitigate this deficiency, [11] incorporates a neighborhood
45 correlation measure in the *inter*-metrics. A spectral matching algorithm working along the lines of [13],
46 which exploits intensity information from the raw underlying image, is proposed in [17]. As [14] argues, [13]
47 suffers also from numerical instability. On the other hand, the approach in [14] performs poorly whenever the
48 features corresponding to the main eigenmodes of either image undergo structural changes. Furthermore, the
49 technique is not robust to point-jitter. [14] can be improved by employing kernel methods in the definition of
50 the *intra*-metrics matrices [15]. This is for instance implemented with a kernel PCA in [21, 22], where the data
51 are embedded into a higher-dimensional space through a Gaussian kernel. Along with not accommodating
52 the presence of underlying images, both the approaches in [13] and in [14] do not incorporate information
53 about features connectivity or features distinctiveness, which the original images may be endowed with –
54 [12] proposes an improvement over the correspondence procedure by incorporating outside information.

55 Other techniques in the literature study variants of the correspondence detection problem. Statistical
56 methods [23] embed probability distributions over point sets. The registration task can then be reframed
57 as a maximum likelihood estimation procedure with coherence constraints [10], or as a distribution align-

58 ment problem [8]. On the other hand, deterministic optimization-based approaches based on minimum
59 least-squares problems [5], or constrained global energy minimization techniques using objective functions
60 quantifying geometric similarity and features coherence [18] have been put forward to tackle the corre-
61 spondence and matching problem. These techniques are intrinsically different in nature than the method
62 discussed in this work in that 1) they possibly embed probabilistic information over the data sets, 2) they do
63 not fully exploit the texture information underlying the images, and 3) they are iterative (as opposed to being
64 one-shot), since they leverage optimization procedures (e.g., energy minimization or likelihood maximiza-
65 tion). As such, while these techniques generally aim at solving the same problem, they are methodologically
66 and structurally different than the approach discussed in the present contribution.

67 The purpose of this work is to propose a novel approach for the general problem of point set matching.
68 With reference to the earlier literature review, the approach 1) leverages and combines the *intra*- and *inter*-
69 information obtained from the pair of feature sets; 2) exploits the use of spectral techniques over the feature
70 sets — the proposed method is non iterative and computationally quite efficient; 3) embeds additional
71 information deriving from the possible knowledge of an existing graphical structure over the single features
72 set, as well as from the presence of underlying raw images. All this heterogeneous information is included
73 in a single method by exploiting structural properties of kernel matrices.

74 As in [13], we make use of a *pairing matrix*, which relates pairs of points taken from the two sets.
75 In addition, we allow the elements of the pairing matrix to depend on a combination of possible metrics,
76 each of which is defined over the two sets of points. This “library” of metrics represents a set of different
77 possible measures that quantify attributes of similarity between the pair of feature sets under study, and
78 can be extended based on the problem under study and the available information. For example, one metric
79 introduces a notion of adjacency between the spatial coordinates of pairs of points, as proposed in [13]; two
80 related metrics compare the eigenmodes between intra-matrices defined for each of the two feature sets, as
81 in [14]; an additional metric exploits the presence of a graphical structure underlying the points in each
82 image; a final metric instead uses the information coming from actual raw images underlying the feature
83 points. Thanks to structural properties of kernel matrices, such measures are combined into a single pairing
84 matrix obtained as the multiplication of the single pairing matrices. This new pairing matrix accommodates
85 different and heterogeneous information: such versatility allows the application of the approach to diverse
86 data sets.

87 The technique performs robustly on a variety of application domains, by automatically adapting the set
88 of useful metrics to the particular case under study. Specifically, the multiplicative structure of the pairing
89 matrix ensures that if a particular metric is not discriminative for a pair of images it is automatically
90 overridden by the other more relevant metrics. The procedure is tested over a few case studies drawn from
91 different domains, in order to fairly assess its performance with regard to the following major characteristic

92 benchmarks: 1) the presence of outliers (i.e., the emergence or the deletion of a subset of the features) in
 93 each image due to structural changes; 2) substantial rotations and translations between the pair of images;
 94 and 3) random position jitter.

95 The contribution unfolds as follows. Section 2 introduces the theoretical concepts and delineates the
 96 main technical aspects underlying the proposed technique. In particular, Sections 2.1 and 2.2 discuss the
 97 notion of pairing matrix and its use for the point set matching problem, Section 2.3 introduces a library
 98 of different possible metrics that can be embedded within the pairing matrix, and Section 2.4 derives the
 99 algorithmic complexity of the proposed approach. Section 3 covers a number of case studies, where the
 100 technique proposed in this work is tested and benchmarked. The examples are both synthetic (Sec. 3.2)
 101 and drawn from biological experimental data (Sec. 3.3), as well as taken from known benchmarks in the
 102 literature ([1], Sec. 3.4). Section 4 concludes the work and delineates future research directions.

103 2. Theory

104 This Section discusses the theory that underpins the proposed point set matching procedure.

105 2.1. The Pairing Matrix for Point Set Matching

106 Consider two sets of points $X = \{x_1, \dots, x_m\}$ and $Y = \{y_1, \dots, y_n\}$, with elements lying in \mathbb{R}^d for some
 107 finite d . As in [13], we introduce a *pairing matrix* $\mathcal{Z} \in \mathbb{R}^{m \times n}$, with entries $z_{ij} \in \mathbb{R}$. Each element z_{ij} is
 108 intended to express a measure of similarity between point $x_i \in X$ and $y_j \in Y$. The definition of the elements
 109 in \mathcal{Z} is elaborated in Sections 2.2 and 2.3. Given the pairing matrix \mathcal{Z} , the selection of pairs of matching
 110 points from the two sets is performed after a normalization procedure, and is described in the following.

111 The matrix \mathcal{Z} is preprocessed by computing its singular-value decomposition

$$\mathcal{Z} = TDU, \tag{1}$$

112 where $D \in \mathbb{R}^{m \times n}$, whereas T, U are properly-sized orthogonal matrices. We replace diagonal elements d_{pp}
 113 of D , $p = 1, 2, \dots, \min\{m, n\}$, with identity constants, which yields

$$\tilde{\mathcal{Z}} = TEU, \tag{2}$$

114 where $e_{ij} = d_{ij}$, $i \neq j$, $i = 1, \dots, m$, $j = 1, \dots, n$, $e_{pp} = 1$, $p = 1, \dots, \min\{m, n\}$. This technique is generally
 115 known as *whitening*. The matrix $\tilde{\mathcal{Z}}$ is orthogonal and is the matrix that maximizes the trace of $\tilde{\mathcal{Z}}^T \mathcal{Z}$ [13].
 116 Furthermore, the largest elements in $\tilde{\mathcal{Z}}$ correspond to candidate matching pairs as follows: the pair (x_i, y_j)
 117 is matched if and only if z_{ij} is the largest element both of row i and of column j . This strong correspondence
 118 implies a “mutual consent” to the match: indeed, if z_{ij} is the largest element of row i but not of column j ,
 119 point x_i is similar to y_j , but not the contrary. As such, the pair (x_i, y_j) is not a valid match.

120 If one considers each row i as an n dimensional vector r_i , then \mathcal{Z} is a map from point x_i into vector r_i .
 121 Ideally, a pairing matrix \mathcal{Z} should be sparse, with a single non-zero element per row and linearly independent
 122 rows. In such a case, r_i coincides with one of the axis and as such is the farthest possible from any other
 123 vector row. More generally, if a row vector r_i is close to the axis $e_j \in \mathbb{R}^n$, then it is likely for the pair
 124 (x_i, y_j) to be a match. However, if two vectors r_{i_1} and r_{i_2} , $i_1, i_2 \in \{1, \dots, m\}$, are adjacent to the same
 125 axis e_j , then the corresponding points x_{i_1} and x_{i_2} “compete” for the match with y_j . Setting the singular
 126 values of \mathcal{Z} to be unitary corresponds to a spatial outspread of its row vectors, which thus alleviates such
 potential conflicts. Figure 1 displays an instance of \mathcal{Z} and $\tilde{\mathcal{Z}}$ matrices corresponding to sets of cardinality

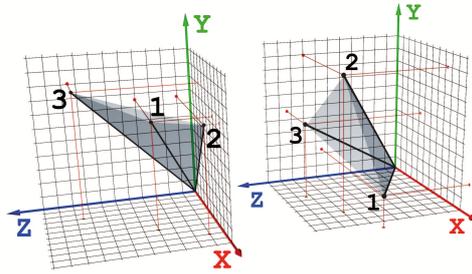


Figure 1: Graphical comparison of pairing matrix \mathcal{Z} (left) and its normalized version $\tilde{\mathcal{Z}}$ (right), where $\mathcal{Z} =$

$$\begin{bmatrix} 0.7000 & 0.7500 & 0.4000 \\ 0.9000 & 0.9500 & 0.1000 \\ 0.3000 & 0.9000 & 0.8500 \end{bmatrix}$$
 and $\tilde{\mathcal{Z}} = \begin{bmatrix} 0.8519 & 0.1527 & 0.5010 \\ 0.4249 & 0.7607 & 0.4907 \\ 0.3062 & 0.6309 & 0.7129 \end{bmatrix}$. Notice that the row vectors r_1, r_2, r_3 of $\tilde{\mathcal{Z}}$ do not cluster
 and better spread in space. The pairing matrix \mathcal{Z} yields one pair (2, 2), whereas $\tilde{\mathcal{Z}}$ yields the pairs (1, 1), (2, 2), (3, 3).

127
 128 In this example where $m = n$, the above operation corresponds to the normalization the volume of
 129 the associated prism.

130 Unlike other iterative approaches [5, 18], the method yields the matching with a single-step calculation.
 131 The formal algorithmic complexity of the approach is derived in Section 2.4.

132 2.2. Flexible Design of the Pairing Matrix

133 As previously mentioned, the elements z_{ij} of \mathcal{Z} represent a measure of similarity between point x_i and
 134 point y_j . A kernel function $\mathcal{K} : X \times Y \rightarrow \mathbb{R}$ is used to define these elements as an inner product in a (possibly
 135 infinite) dimensional space [15], so that

$$z_{ij} = \mathcal{K}(x_i, y_j) = \langle \phi(x_i), \phi(y_j) \rangle \quad (3)$$

136 and

$$\mathcal{Z} = \begin{bmatrix} z_{11} & \dots & z_{1n} \\ \vdots & \ddots & \vdots \\ z_{m1} & \dots & z_{mn} \end{bmatrix} = \begin{bmatrix} \mathcal{K}(x_1, y_1) & \dots & \mathcal{K}(x_1, y_n) \\ \vdots & \ddots & \vdots \\ \mathcal{K}(x_m, y_1) & \dots & \mathcal{K}(x_m, y_n) \end{bmatrix}. \quad (4)$$

137 As a special case of (3), one can select $\mathcal{K}(x_i, y_j) = \langle x_i, y_j \rangle$, whereby $\phi(\cdot) = \text{id}(\cdot)$ is the identity function. In
 138 general, given a proper kernel function \mathcal{K} , Mercer's Theorem [15, Thm. 3.13] guarantees the existence of
 139 the embedding function $\phi(\cdot)$, though it may not be explicitly known.

140 Given a (possibly infinite) set of kernel functions $\mathcal{K}_1, \dots, \mathcal{K}_t, t \in \mathbb{N}$, the closure property [15, Prop. 3.22]
 141 warrants that

$$\mathcal{K}^s(x_i, y_j) = \sum_{k=1}^t \mathcal{K}_k(x_i, y_j) \quad \text{and} \quad \mathcal{K}^p(x_i, y_j) = \prod_{k=1}^t \mathcal{K}_k(x_i, y_j) \quad (5)$$

142 are also kernel functions. Hence, by considering pairing matrices $\mathcal{Z}_1, \dots, \mathcal{Z}_t$ built over the kernel functions
 143 $\mathcal{K}_1, \dots, \mathcal{K}_t$ as in (4), the new matrices

$$\mathcal{Z}^s = \sum_{k=1}^t \mathcal{Z}_k \quad \text{and} \quad \mathcal{Z}^p = \prod_{k=1}^t \mathcal{Z}_k \quad (6)$$

144 are still valid kernel-based pairing matrices. (Here the symbols \sum and \prod represent point-wise sum and
 145 multiplication.)

146 For any legitimate choice of the kernel function, the pairing matrix \mathcal{Z} relates to the notion of Gram
 147 matrix [15], which connects with a wealth of literature on kernel methods for pattern analysis. However,
 148 since in the present instance \mathcal{Z} is applied over pairs of points extracted from two different sets (rather than
 149 from the very same set), it does not formally belong to this class of matrices.

150 The kernels \mathcal{K}_k are built around metrics, or distances d_k , which are suitable for the matching problem.
 151 We select a Gaussian as the kernel function:

$$\mathcal{K}_k(x_i, y_j) = e^{-\frac{d_k^2(x_i, y_j)}{\sigma_k^2}}, \quad (7)$$

152 where σ_k is a parameter of choice that controls the degree of interaction between the two feature points.
 153 Next, we introduce a library of possible metrics, which we will apply in a number of heterogeneous case
 154 studies.

155 2.3. A Library of Metrics

156 Let us consider the following set of metrics $d_k : X \times Y \rightarrow \mathbb{R}, k = 1, \dots, 5$:

Metric	Definition
$d_1(x_i, y_j)$	$\ x_i - y_j\ _p, p > 0$
$d_2(x_i, y_j)$	$\cos(\mathfrak{m}(i), \mathfrak{m}(j))$
$d_3(x_i, y_j)$	$\cos(\tilde{\mathfrak{m}}(i), \tilde{\mathfrak{m}}(j))$
$d_4(x_i, y_j)$	$ \mathfrak{d}(x_i) - \mathfrak{d}(y_j) $
$d_5(x_i, y_j)$	$\mathfrak{t}\mathfrak{d}(x_i, y_j)$

157

158 The metrics are further discussed in the following.

- 159 1. d_1 is a p -norm distance between pairs of points taken from the two sets. The present case studies
160 consider the Euclidean norm ($p = 2$).
- 161 2. d_2 is a measure of the distance of the i -th *mode* $\mathbf{m}(i)$ and the j -th *mode* $\mathbf{m}(j)$ associated to each
162 of the two sets. As suggested in [14], the mode \mathbf{m} of a point set is computed as follows: first, a
163 square proximity matrix defined according to the intra-distances between the features of the image is
164 introduced; it is successively diagonalized and its first $\min\{m, n\}$ eigenvectors, sorted according to the
165 largest eigenvalues, are regarded as its modes. The distance between the modes $\mathbf{m}(i)$ and $\mathbf{m}(j)$ of the
166 two graphs is then computed as their cosine.
- 167 3. This metric is valid if the sets X, Y are embedded with a graphical structure. The distance d_3 is
168 characterized analogously as d_2 , however the modes $\tilde{\mathbf{m}}$ are computed by considering a proximity matrix
169 that has non-zero elements only if the corresponding pair of vertices are connected by an existing edge
170 in the graph.
- 171 4. The metric d_4 is defined as the absolute value of the difference between the graphical degree of a pair
172 of points, taken respectively from X and from Y . The function \mathbf{d} is the degree of a node in a graph.
173 As for d_3 , this metric is valid if the sets X, Y are embedded with a graphical structure. However, if no
174 graphical structure is present over the sets X and Y , then one such graph can be induced artificially:
175 for instance, edges can be created between pairs of points if their distance is less than an adjustable
176 threshold.
5. Given an image underlying the point sets, the metric d_5 , defined by a function \mathbf{td} , computes the *texture*
difference between a neighborhood of $x_i \in X$ and one of $y_j \in Y$. Here the points x_i, y_j are intended
as features of the corresponding images.

More formally, let us consider the finite discrete domain $L \subset \mathbb{Z}^2$ made up of the pixels of the two-dimensional image frame. Given a point $z \in \mathbb{R}^2$, we define a neighborhood $\mathcal{N}(z, \delta) \subset \mathbb{Z}^2$ as the set of pixels of the image with centroid lying within a radius $\delta > 0$ from z . A function $\mathcal{I} : L \rightarrow \mathbb{R}^+$ specifies the *intensity* of the image over its domain.

Let us now consider the two images L_x, L_y underlying X, Y . Given a radius δ of choice (the choice of δ may be related to σ_5 in (7)), the function $\mathbf{td}(z_1, z_2)$ computes the absolute value of the difference between the intensities of the pixels in the neighborhoods of points z_1 and z_2 :

$$\mathbf{td}(z_1, z_2) = \sum_{p_1 \in \mathcal{N}(z_1, \delta) \cap L_x} \sum_{p_2 \in \mathcal{N}(z_2, \delta) \cap L_y} |\mathcal{I}(p_1) - \mathcal{I}(p_2)|.$$

177 This approach is related to a similar procedure used in [17].

178 The metrics d_1, \dots, d_5 , which are used in kernel functions (7), are then combined through a product into a
179 pairing matrix \mathcal{Z} , as in (5), to be employed in the matching procedure. The multiplicative structure of the
180 pairing matrix ensures that if a particular metric is not discriminative for a pair of images it is automatically

181 overridden by other more relevant metrics. This versatile feature allows the automatic adaptation of the set
 182 of useful metrics to the particular case under study.

183 2.4. Algorithmic Complexity

184 Let us start assuming that we are given t kernel functions $\mathcal{K}_k, k = 1, \dots, t$, as in (7), which are used in
 185 the matching procedure. The procedure is made up of four sequential steps that determine its algorithmic
 186 complexity.

187 Firstly, we build \mathcal{Z} as a point-wise multiplication of the t kernel functions $\mathcal{K}_k, k = 1, \dots, t$, according to
 188 (6). This involves $d \cdot t \cdot m \cdot n$ operations, where m and n are the number of features respectively in X and Y ,
 189 and d is the dimensionality of their spatial component. We expect that d, t are constants that are smaller
 190 than m, n , which results in a $\mathcal{O}(nm)$ computational complexity.

191 Secondly, we compute $\tilde{\mathcal{Z}}$ using a singular value decomposition procedure as in [6]: accordingly, the
 192 computation of U, V and D requires $4m^2n + 8mn^2 + 9n^3$ operations.

193 Thereafter, the whitening procedure is completed by replacing the diagonal elements of D , which easily
 194 results in $\mathcal{O}(\min\{m, n\})$ computational complexity.

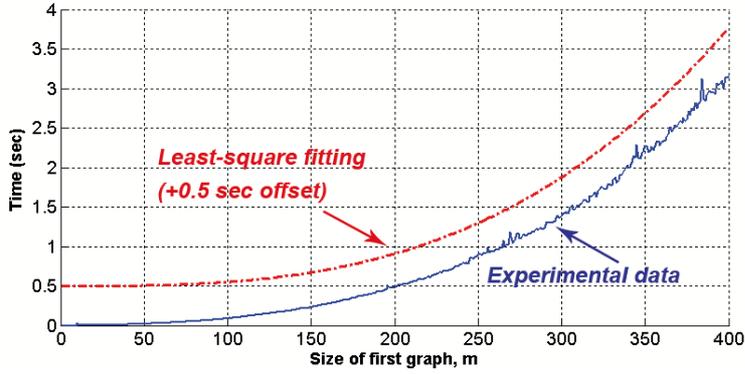
195 Finally, the matching procedure on $\tilde{\mathcal{Z}}$ requires finding the minimum of each row and then scanning the
 196 corresponding column to confirm or discard the matching. This entails in $\binom{\min\{m, n\}}{2}$ operations, which is
 197 $\mathcal{O}((\min\{m, n\})^2)$.

198 The overall computational complexity thus easily add up being $\mathcal{O}(m^2n + mn^2 + n^3)$. Intuitively, we can
 199 state that the computation cost grows with the third power of the size of $|X|$ or $|Y|$.

200 In practice, we have found the computational cost not to be a burden, even using whitening. All the case
 201 studies described in Section 3, which are by no means of trivial dimensionality or structure, have resulted in
 202 real-time executions. In other words, whilst the computed polynomial complexity holds, we have experienced
 203 that the hidden constants in $\mathcal{O}(m^2n + mn^2 + n^3)$ to be quite limited in value. For instance, Table 1 shows the
 204 computational complexity of the matching procedure for the synthetic graph matching test case in Section
 205 3.2, run using $\mathbb{P}_e, \mathbb{P}_v$ and $\mathbb{M}_v = 15.00\%$: for this test, we have found that the complexity $\mathcal{O}(m^2n + mn^2 + n^3)$
 206 can be further computed to be equal to $2.1 \cdot 10^{-7}m^3$, where m is the size of the first graph (here $n \leq m$). On
 207 the side, notice that one can implement a much faster algorithm, albeit at a loss in reliability, by avoiding
 208 the whitening of \mathcal{Z} (second and third step above). This obtains a computational cost of $\mathcal{O}(mn)$.

209 3. Case Studies

210 This Section develops a number of case studies to test the methodology introduced in Section 2. We
 211 show that the particular instance under study dictates what subset of metrics mostly influence the pairing
 212 matrix utilized for the matching procedure and optimize its outcomes. The software and the hardware for
 213 all the case studies are specified in Table 1.



Graph size, m	Computational Time
50 nodes	0.022 sec
75 nodes	0.047 sec
100 nodes	0.089 sec
125 nodes	0.154 sec
150 nodes	0.231 sec
175 nodes	0.347 sec
200 nodes	0.472 sec
250 nodes	0.894 sec
300 nodes	1.396 sec
350 nodes	2.277 sec
400 nodes	3.157 sec

Table 1: Computational complexity for the synthetic graph test case of Section 3.2, tested on a Linux Ubuntu machine, AMD Mobile Sempron 3600+ processor with 2GB memory. The test was built in Matlab r2009b[©]. To reduce noise, for each size m of the graph we run 10 trials and averaged the outcomes. The table shows the experimental result (blue) as well the least square analytical fitting (red), obtained with a multiplicative constant $2.1 \cdot 10^{-7}$ and here displayed with a 0.5 sec offset.

3.1. Performance Assessment of the Point Set Matching Procedure

Consider the two sets of feature points X and Y , with cardinality m and n respectively. The outcomes of the point set matching procedure is compared with the ground truth. The ground truth is known for the synthetic case studies, whereas for the case studies based on real images it is directly assessed over the data sets by independent and unbiased observers. Let us introduce the following four entities:

1. $X_{tm} \subseteq X, Y_{tm} \subseteq Y$ are the two sets of feature points that are correctly matched, of cardinality respectively m_{tm}, n_{tm}
2. $X_{ts} \subseteq X, Y_{ts} \subseteq Y$ are the two sets of feature points that are correctly left un-matched, of cardinality respectively m_{ts}, n_{ts}
3. $X_{fm} \subseteq X, Y_{fm} \subseteq Y$ are the two sets of feature points that are wrongly matched, of cardinality respectively m_{fm}, n_{fm}
4. $X_{fs} \subseteq X, Y_{fs} \subseteq Y$ are the two sets of feature points that are wrongly left un-matched, of cardinality respectively m_{fs}, n_{fs}

Notice, as intuitive, that $m_{tm} + m_{ts} + m_{fm} + m_{fs} = m$ and that $n_{tm} + n_{ts} + n_{fm} + n_{fs} = n$. We define as percentages, over both sets of points, the following quantities:

1. *true matches*, as the ratio of feature points that are correctly matched, i.e. $\frac{m_{tm} + n_{tm}}{m + n}$
2. *true singles*, as the ratio of feature points that are correctly left un-matched, i.e. $\frac{m_{ts} + n_{ts}}{m + n}$
3. *false matches*, as the ratio of feature points that are wrongly matched, i.e. $\frac{m_{fm} + n_{fm}}{m + n}$
4. *false singles*, as the ratio of feature points that are wrongly left un-matched, i.e. $\frac{m_{fs} + n_{fs}}{m + n}$

The outcome of the case studies will be evaluated according to the introduced quality measures.

234 *3.2. Synthetic Graph Matching*

235 We consider a graphical structure $G = (V, E)$ with two dimensional spatial components, which are
 236 constrained to lie within the unit square in the positive quadrant $[0, 1]^2$. The cardinality of the set of
 237 nodes V is equal to 50 and their spatial components are generated uniformly at random within the specified
 238 domain. The edge set E is created between pairs of nodes in V according to a Bernoulli distribution with
 239 mean equal to 0.5, however the edges that are longer than a specified threshold (0.25) are discarded. Figure
 240 2 shows an example of a graph.

241 The graph is then morphed into a new structure $\tilde{G} = (\tilde{V}, \tilde{E})$, according to the following procedure:

- 242 1. The set $\tilde{E} \subseteq E$ is generated from E by discarding each edge according to a Bernoulli probability
 243 distribution with mean \mathbb{P}_e ;
- 244 2. The set $\tilde{V} \subseteq V$ is generated from V by discarding each edge according to a Bernoulli probability
 245 distribution with mean \mathbb{P}_v , and additionally by eliminating the residual edges that connect to vertices
 246 in $E \setminus \tilde{E}$;
- 247 3. The spatial components associated to the elements in \tilde{E} are obtained from those belonging to the cor-
 248 responding elements in E by perturbing them with the addition of a random variable that is uniformly
 249 distributed within the square $\frac{\mathbb{M}_v}{2}[-1, 1]$. In other words, the original coordinates are subjected to a
 250 uniform perturbation that amounts to $\mathbb{M}_v\%$ of their maximum possible value.

251 The matching procedure proposed in Section 2 is tested on a cohort of pairs of graphs (G, \tilde{G}) , parameter-
 252 ized by the input configuration $(\mathbb{P}_e, \mathbb{P}_v, \mathbb{M}_v)$ used to generate them: for each combination $(\mathbb{P}_e, \mathbb{P}_v, \mathbb{M}_v)$ of
 253 perturbation parameters we average the outcomes of 2000 simulations.

254 Figures 2 and 3 represent respectively a single pair of test graphs $(\mathbb{P}_e = \mathbb{P}_v = \mathbb{M}_v = 15.00\%)$, and the
 255 outcomes of the matching procedure.

256 We have employed the first four metrics d_1, d_2, d_3 and d_4 . The metric d_5 is not employed, since the
 257 artificial graphs have no underlying physical image that can be exploited for the matching procedure. Table
 258 2 reports the results for each configuration of the perturbation parameters $(\mathbb{P}_e, \mathbb{P}_v, \mathbb{M}_v)$ as the average of the
 259 2000 simulations. The experiments are divided in five batches: in the first, we uniformly modify the three
 260 perturbation parameters; in the following four, we fix two of the three parameters and modify the remaining
 261 one by using values that match those of the first batch of experiments. The monotonicity of the performance
 262 outputs of the algorithm with respect to the perturbation level provides evidence of the consistency of the
 263 procedure (see first set of simulations). The results of the last four groups of simulations “lie within” those
 264 of the first batch (the comparison ought to be done by looking at the accrued percentages for true and false
 265 value pairs). By comparing the third result of each group of experiment with the first two, one can observe
 266 that the elimination of edges or vertices affects the quality of the outcomes more than the perturbation
 267 of the spatial coordinates of the vertices. This is despite the fact that spatial perturbations can result in

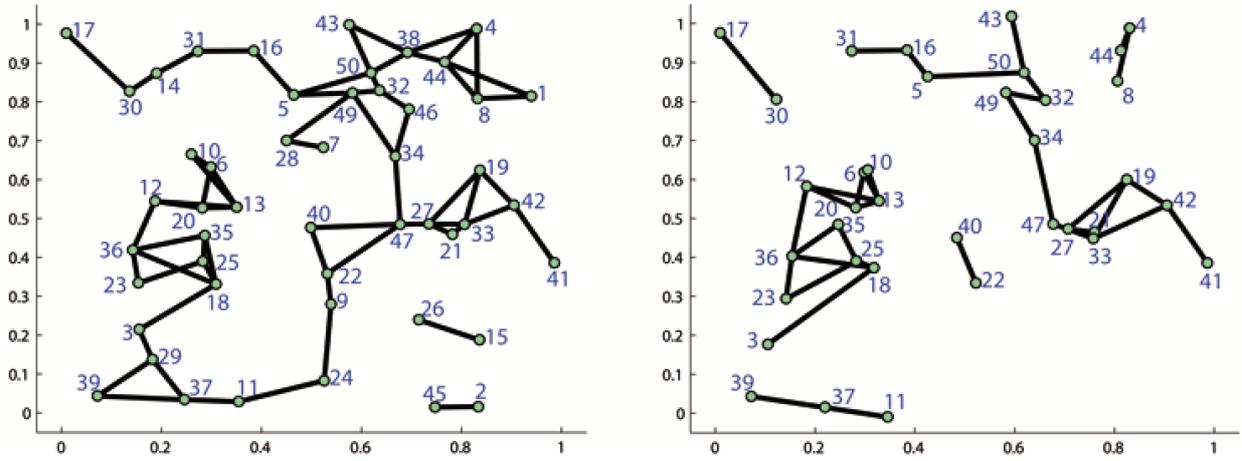


Figure 2: Synthetic graph generation and perturbation. The original graph (left) contains 50 nodes. The perturbed graph (right) was generated using $\mathbb{P}_e = \mathbb{P}_v = M_v = 15.00\%$. The blue labels mark the nodes and provide the ground truth correspondence.

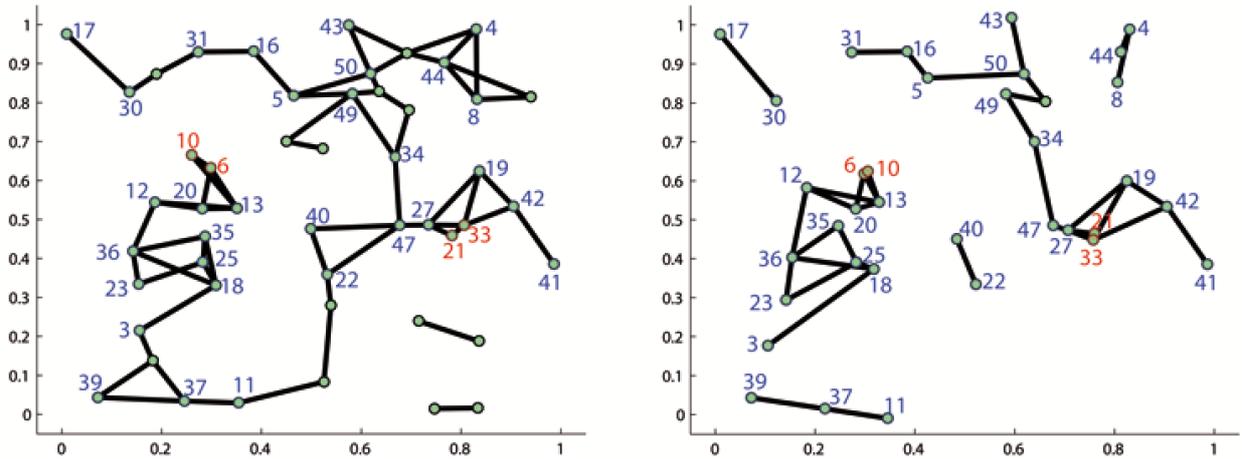


Figure 3: Outcomes of the matching procedure for the synthetic graphs of Figure 2. Here the blue labels indicate correctly matched nodes (true matches), whereas red labels shows the wrongly matched ones. Nodes without labels are correct single nodes.

268 feature crossover. Furthermore, as intuitive, the elimination of an edge (first result of the last four groups)
 269 affects the results more than that of a vertex (second result).

Perturbation			Output Performance			
\mathbb{P}_e	\mathbb{P}_v	\mathbb{M}_v	true matches	true singles	false matches	false singles
15.00%	15.00%	15.00%	71.01%	7.50%	19.37%	3.12%
12.00%	12.00%	12.00%	79.83%	6.66%	12.07%	1.44%
9.00%	9.00%	9.00%	87.65%	4.37%	7.10%	0.88%
6.00%	6.00%	6.00%	93.06%	3.38%	3.36%	0.20%
3.00%	3.00%	3.00%	97.09%	1.82%	1.04%	0.05%
12.00%	15.00%	15.00%	72.40%	7.21%	18.59%	2.80%
15.00%	12.00%	15.00%	73.17%	6.05%	18.64%	2.14%
15.00%	15.00%	12.00%	78.07%	8.94%	12.19%	1.90%
9.00%	12.00%	12.00%	80.29%	6.71%	11.41%	1.59%
12.00%	9.00%	12.00%	82.31%	4.32%	12.05%	1.32%
12.00%	12.00%	9.00%	84.49%	6.64%	7.80%	1.07%
6.00%	9.00%	9.00%	87.86%	5.62%	6.03%	0.49%
9.00%	6.00%	9.00%	89.31%	3.32%	6.72%	0.65%
9.00%	9.00%	6.00%	90.86%	5.23%	3.42%	0.59%
3.00%	6.00%	6.00%	93.26%	3.60%	2.78%	0.36%
6.00%	3.00%	6.00%	95.17%	1.68%	2.99%	0.16%
6.00%	6.00%	3.00%	94.86%	3.70%	1.20%	0.24%

Table 2: Outcomes of the matching procedure tested on sets of randomly generated and successively perturbed graphs. We have run 2000 simulation for each configuration of perturbation parameters and reported the average of the outcomes. For each of the 2000 simulations we have first generated a graph, then perturbed it. The perturbation level is tuned via three parameters: \mathbb{P}_e , the probability that an edge is erased from the original graph; \mathbb{P}_v , the probability that a vertex is eliminated from the original graph; \mathbb{M}_v , the level of spatial perturbation applied to a vertex of the original graph.

270 3.3. Image Registration of Biological Data: The *Drosophila* Wing Case Study

271 This experimental study aims at matching two network structures extracted from biological data.¹ Each
272 network describes the cellular epithelium of a wing of *Drosophila melanogaster*, the common fruit fly. The
273 experimental data are obtained with confocal microscopy techniques a few hours after puparium formation.
274 It is of interest for the developmental biologist to have access to quantitative data relating to the network
275 structure of the epithelium. The graphical structure is extracted from single frames that belong to time-
276 lapse movies of the epithelium. The details of the computer vision technique used to extract the network
277 from a single frame are formally presented in [16]. Along with the collection of the graphical structures
278 corresponding to each frame, it is important to match the networks extracted from pairs of frames that are

¹The images have been provided by the Axelrod Lab, at the Department of Pathology, Stanford University School of Medicine, Stanford, USA. Members of the Lab have also contributed in the interpretation of the outcomes of the registration procedure.

279 successive in time. This procedure is also known as the *registration* of the frames of the movie.

280 The experimental data consist of 50 frames consecutive in time, corresponding to 49 pairs of images.
Figure 4 shows frames (frame 22 and 23) taken from the wet lab experimental data.

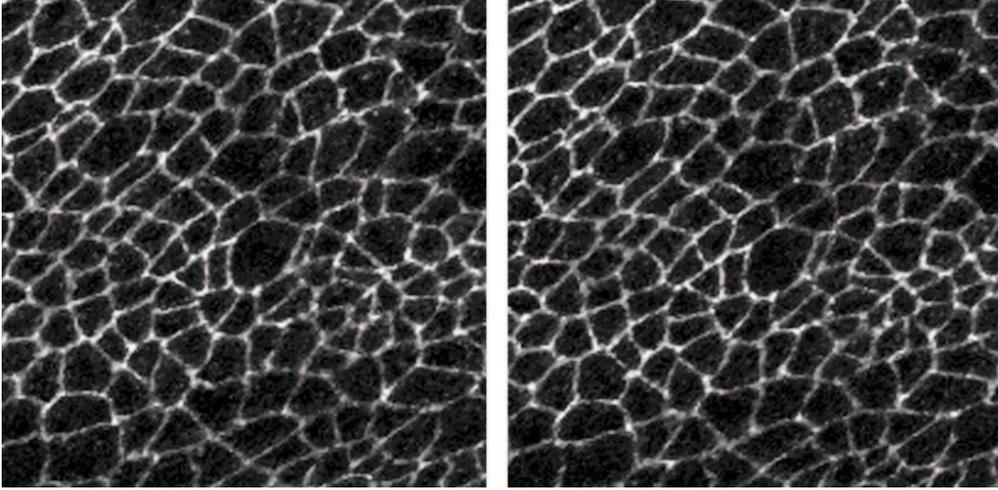


Figure 4: Frames 22 and 23 considered for the matching procedure. The images are part of a 40 frame movie and refer to a section of the epithelium of the *Drosophila melanogaster* wing. The polygonal structures are 2-d sections of the epithelial cells.

281
282 For the instance under study, we have employed the metrics d_1 and d_5 . The use of d_5 is dictated by
283 the availability of an actual image containing meaningful information for the matching. The *intra-metrics*
284 d_2, d_3 and d_4 are discarded, which is explained by observing the similarity of the neighborhood structure
285 for most of the nodes in the graph. In other words, if most of the internal nodes have a similar number of
286 connected edges, then the information provided by the metric d_4 is redundant. Similar considerations hold
287 for the metrics d_2, d_3 .

288 Figure 5 displays the graphical structures extracted from the frames in Figure 4, and labeled with the
289 outcome of the matching procedure. Unlike the study in Section 3.2, the ground truth has been provided
290 by manual observation from an independent and unbiased observer. Table 3 displays the outcomes of the
291 procedure, averaged over the 49 tests. The results appear to be rather good when compared to others in
292 the biology literature [9], especially given the complexity of the structures and of the dynamics under study
293 (cells both appear to divide and to exit the epithelium, the frames are subject to translation, and the
294 images are quite noisy — the last two are known issues for algorithms as in [13]).

295 3.4. Point Set Matching over a Literature Benchmark: The “CMU House”

296 We have finally tested our procedure on a known benchmark from the computer vision literature, known
297 as the “CMU House” [1]. This benchmark contains a set of 110 pictures of a toy house, taken over a black
298 background. We have extracted a set of features from each image by applying a corner detector [24]. The

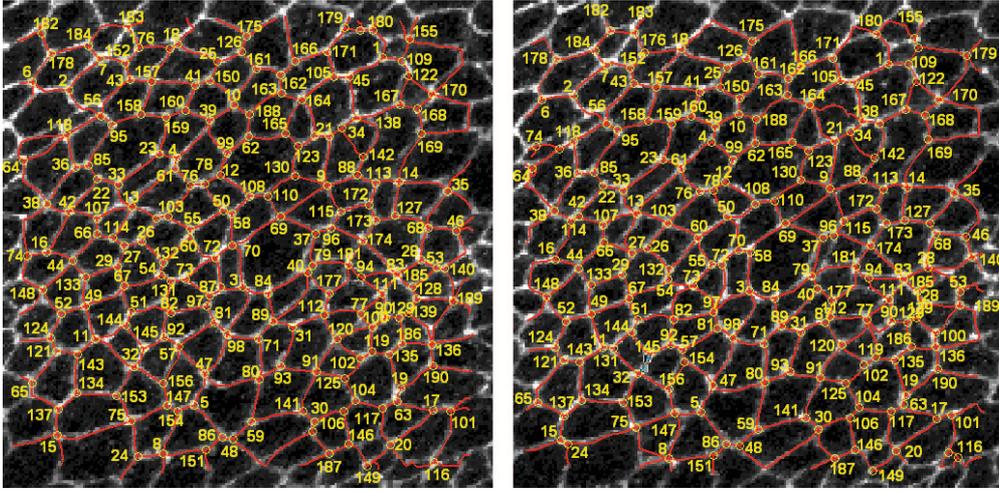


Figure 5: The matching procedure applied to the networks of Figure 4. The yellow labels over the nodes correspond to matches (both correct and wrong ones). The unlabelled nodes are single nodes (both correct and wrong ones).

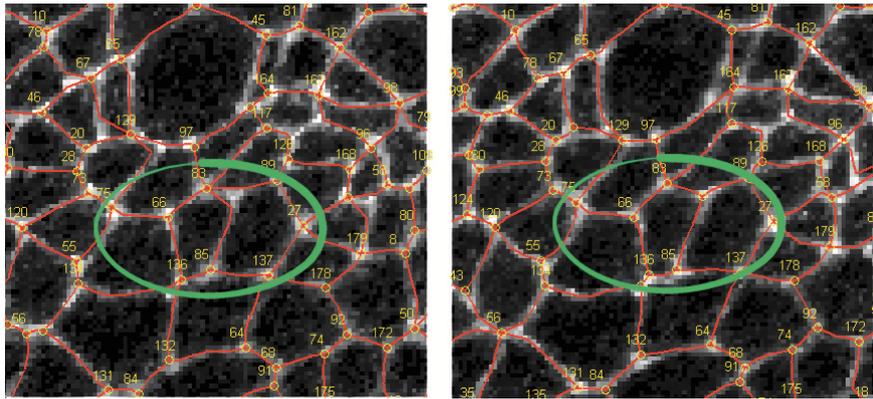


Figure 6: A particular of the matching procedure from Figure 5, involving a topological change. The green circle highlights a difficult match that is correctly resolved.

Output Performance			
true matches	true singles	false matches	false singles
88.23%	3.61%	7.92%	0.24%

Table 3: Outcomes of the matching procedure tested on 49 pairs of networks extracted from 50 successive frames of a movie. The results are averages over the 49 tests. The movie refers to the morphogenesis and the dynamics of a section of the wing of *Drosophila melanogaster*.

299 obtained sets have a cardinality that is very similar to the sets used in [4, 22] for the same benchmark, which
 300 leads to a fair comparison with those results.

301 We have tested our algorithm on two experimental setups. Firstly, we have matched the 109 sequential
 302 pairs of images (1-2, 2-3, \dots , 109-110). Figure 7 displays the output of one such pairing: the green labels are
 303 obtained from the matching procedure. Secondly, we have matched 109 pairs of distinct images, randomly
 304 chosen from the set. Figure 8 displays the output of one such matching: in green are the correct labels
 305 obtained from the matching procedure, whereas in red are the wrong outcomes.

306 The first study is meant to test the robustness of the method with respect to positional jitter, while the
 307 second targets the performance against large transformations and the presence of feature occlusions. (Both
 308 studies focus on known issues reported for algorithms as in [13].) Table 4 displays the results as averages
 309 over the 109 tests. The outcomes of both studies appear to sensibly improve those in [4], and to remarkably
 310 improve those in [22]. Notice that the performance measure in [4, 22] is based exclusively on the second
 311 component of the pair of images, and hence slightly differs from the one used in this work, which we believe
 312 is more accurate. Also, the statistics in both [4, 22] are quite limited in sample size and image range.

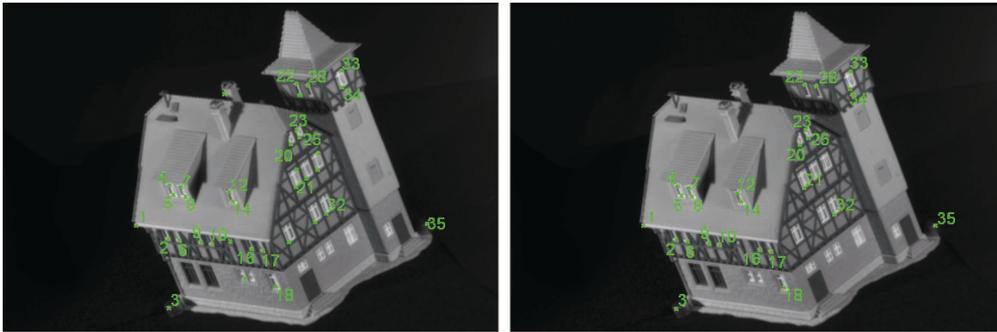


Figure 7: Two successive images (frames 1 and 2) from the CMU House benchmark [1]. The green labels are obtained from the matching procedure. The outcome is in this case perfect.

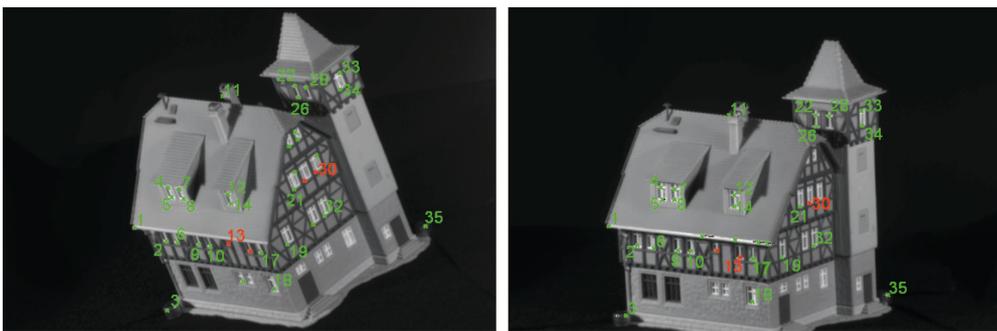


Figure 8: Two random images (frames 1 and 67) considered for the matching procedure over the CMU House benchmark [1]. The green labels correspond to correctly matched points (true matches and true singles), whereas the red labels mark wrong outcomes (false matches and false singles)

Input	Output Performance			
image pairs	true matches	true singles	false matches	false singles
sequential	93.32%	5.48%	0.73%	0.47%
random	75.86%	16.20%	5.68%	2.26%

Table 4: Outcomes of the matching procedure tested on 109 pairs of feature sets extracted from 110 images in [1]. The results are averages over the 109 tests. The top line refers to the 109 pairs of sequential images, whereas the bottom one to 109 pairs of randomly extracted images.

4. Conclusions and Future Work

This article has proposed a versatile technique to perform point set matching over features extracted from images. The approach combines a number of ideas from related approaches in the literature. Its overall flexibility results from the possibility to define a library of metrics, from which similarity measures are selected and later employed over the specific matching problem. These heterogeneous measures are combined into a single pairing matrix, which is then manipulated via spectral techniques to obtain the actual matching.

The method has been tested on a number of different experimental studies, which have highlighted its performance, its robustness, as well as its computational scalability.

It is of future interest to come up with novel, descriptive metrics that can extend the applicability of the library and its usefulness to new domains of study.

References

- [1] <http://vasc.ri.cmu.edu/idb/html/motion/house/index.html>.
- [2] S. S. Beauchemin and J. L. Barron. The computation of optical flow. *ACM Computing Surveys (CSUR)*, 27:433–466, 1995.
- [3] V.D. Blondel, A. Gajardo, M. Heymans, P. Senellart, and P. van Dooren. A measure of similarity between graph vertices: Applications to synonym extraction and web searching. *SIAM Review*, 46(4):647–666, 2004.
- [4] M. Carcassoni and E.R. Hancock. Correspondence matching with modal clusters. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(12):1609–1615, 2003.
- [5] H. Chui and A. Rangarajan. A new point matching algorithm for non-rigid registration. *Computer Vision and Image Understanding*, 89(2-3):114–141, 2003.
- [6] G. Golub and C. Reinsch. Singular value decomposition and least squares solutions. *Numerische Mathematik*, 14(5):403–420, April 1970.
- [7] D.P. Huttenlocher and W.J. Rucklidge. A multi-resolution technique for comparing images using the Hausdorff distance. In *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR 93)*, pages 705–706, Jun 1993.
- [8] B. Jain and B.C. Vemuri. A robust algorithm for point set registration using mixture of gaussians. In *Proceedings of the International Conference on Computer Vision (ICCV 05)*, pages 1246–1251, 2005.

- 340 [9] D. Ma, K. Amonlirdviman, R. Raffard, A. Abate, C.J. Tomlin, and J.D. Axelrod. Cell packing influences planar cell
341 polarity signaling. *The Proceedings of the National Academy of Sciences*, 105(48):18800–18805, December 2008.
- 342 [10] A. Myronenko, X. Song, and M. Carreira-Perpinan. Non-rigid point set registration: Coherent point drift. In B. Schölkopf,
343 J. Platt, and T. Hoffman, editors, *Advances in Neural Information Processing Systems (NIPS) 19*, pages 1009–1016. MIT
344 Press, Cambridge, MA, 2007.
- 345 [11] M. Pilu. A direct method for stereo correspondence based on singular value decomposition. In *Proceedings of the Computer
346 Vision and Pattern Recognition Conference (CVPR 97)*, pages 261–266, Jun 1997.
- 347 [12] S. Sclaroff and A.P. Pentland. Modal matching for correspondence and recognition. *IEEE Transactions on Pattern
348 Analysis and Machine Intelligence*, 17(6):545–561, 1995.
- 349 [13] G.L. Scott and H.C. Longuet-Higgins. An algorithm for associating the features of two images. *Proceedings of the Royal
350 Society London*, B-244:21–26, 1991.
- 351 [14] L.S. Shapiro and J.M. Brady. Feature-based correspondence: An eigenvector approach. *Image and Vision Computing*,
352 10(5):283–288, June 1992.
- 353 [15] J. Shawe-Taylor and N. Cristianini. *Kernel Methods for Pattern Analysis*. Cambridge University Press, New York, NY,
354 USA, 2004.
- 355 [16] A. Silletti, A. Cenedese, and A. Abate. The emergent structure of the *Drosophila* wing: a dynamic model generator. In
356 *Proceedings of the International Conference on Computer Vision, Theory and Applications (VISAPP 09)*, pages 406–410,
357 Lisboa, PT, February 2009.
- 358 [17] S.H. Srinivasan and M. Kankanhalli. Wide baseline spectral matching. In *Proceedings of the 2003 International Conference
359 on Multimedia and Expo (ICME 03)*, pages 93–96, Washington, DC, USA, 2003.
- 360 [18] L. Torresani, V. Kolmogorov, and C. Rother. Feature correspondence via graph matching: Models and global optimization.
361 In *Proceedings of the European Conference on Computer Vision (ECCV 08)*, pages 596–609, 2008.
- 362 [19] S. Ullman. *The Interpretation of Visual Motion*. MIT Press, Cambridge, MA, USA, 1979.
- 363 [20] S. Umeyama. An eigendecomposition approach to weighted graph matching problems. *IEEE Transactions on Pattern
364 Analysis and Machine Intelligence*, 10(5):695–703, 1988.
- 365 [21] H.F. Wang and E.R. Hancock. A kernel view of spectral point pattern matching. In *Structural, Syntactic, and Statistical
366 Pattern Recognition*, volume 3138 of *Lecture Notes in Computer Science*, pages 361–369. Springer Verlag, 2004.
- 367 [22] H.F. Wang and E.R. Hancock. Correspondence matching using kernel principal components analysis and label consistency
368 constraints. *Pattern Recognition*, 39(6):1012–1025, 2006.
- 369 [23] W.M. Wells. Statistical approaches to feature-based object recognition. *International Journal of Computer Vision*,
370 21(1-2):63–98, 1997.
- 371 [24] C.H. Xiao and H. C.Y. Nelson. Corner detector based on global and local curvature properties. *Optical Engineering*,
372 47(5):057008,1–12, 2008.