J Math Imaging Vis
DOI 10.1007/s10851-014-0546-8

Globally Optimal Joint Image Segmentation and Shape Matching Based on Wasserstein Modes

Bernhard Schmitzer · **Christoph Schnörr**

Received: 11 July 2014 / Accepted: 29 October 2014

Abstract A functional for joint variational object segmentation and shape matching is developed. The formulation is based on optimal transport w.r.t. geometric distance and local feature similarity. Geometric invariance and modelling of object-typical statistical variations is achieved by introducing degrees of freedom that describe transformations and deformations of the shape template. The shape model is mathematically equivalent to contour-based approaches but inference can be performed without conversion between the contour and region representations, allowing combination with other convex segmentation approaches and simplifying optimization. While the overall functional is non-convex, non-convexity is confined to a low-dimensional variable. We propose a locally optimal alternating optimization scheme and a globally optimal branch and bound scheme, based on adaptive convex relaxation. Combining both methods allows to eliminate the delicate initialization problem inherent to many contour based approaches while remaining computationally practical. The properties of the functional, its ability to adapt to a wide range of input data structures and the different optimization schemes are illustrated and compared by numerical experiments.

Keywords Image segmentation · Object matching · Optimal transport · Shape analysis · Convex relaxation

B. Schmitzer (✉) · C. Schnörr
University of Heidelberg, Heidelberg, Germany
e-mail: Bernhard.Schmitzer@iwr.uni-heidelberg.de

1 Introduction

1.1 Motivation

Object segmentation and matching are fundamental problems in image processing and computer vision as they form the basis for many high-level approaches to understanding an image. They are intimately related: segmentation of the foreground is a prerequisite for matching in a sequential processing pipeline. Whereas, when performed simultaneously, matching with a template of the sought-after object (e.g. starfish, car, etc.) as prior knowledge can help guiding segmentation to become more robust to corruption of local image features through noise, occlusion and other distortions. Naturally the combined problem is more complicated.

Today convex variational methods can solve image labelling and segmentation problems based on local cues exactly or in good approximation. But combining an object segmentation functional with a shape prior entails a delicate tradeoff between descriptive power and computational complexity. Sophisticated shape priors are often described by highly non-convex functionals whereas convex shape prior functionals tend to be rather simplistic. Incompatibility between different shape representations within one approach or the requirement of geometric invariance are common causes of difficulty.

In this paper we present a shape prior functional for simultaneous object segmentation and matching which has been designed specifically to address the issues of representation incompatibility and geometric invariance. Using optimal transport and the differential geometric structure of the 2-Wasserstein space for regularization, one can combine appearance modelling, description of statistical shape variations and geometric invariance in a mathematically uniform way. The linear programming formulation of opti-

Published online: 07 November 2014

mal transport due to Kantorovich allows for an adaptive convex relaxation which can be used for a globally optimal branch and bound scheme, thus avoiding the initialization problem from which most non-convex approaches suffer.

1.2 Related Literature

Image Segmentation and Shape Priors. Variational methods based on convex relaxations have been successfully applied to obtain globally optimal (approximate) solutions to the originally combinatorial image labelling or segmentation problem [8,22,29]. The segmentation is usually encoded by (relaxed) indicator functions which allow for simple and convex formulation of *local* data matching terms and regularizers that encourage *local* boundary regularity, such as total variation and its generalizations. However, introducing *global* regularizers, such as shape priors, into such models is difficult. Convex shape priors based on indicator functions are conceivable but tend to be rather simplistic and lack important features such as geometric invariance [18,30].

Somewhat complimentary is the representation of shapes by their outline contours. Treated as infinite dimensional manifolds [26,34,38] such representations can be used to construct sophisticated shape modelling functionals [10,12]. But matching contours with local image data usually yields non-convex functionals that can only be optimized locally via gradient descent. Often one has to internally convert the contour to the region representation. Therefore such approaches require a good initialization to yield reasonable results.

Object Registration. Independent of the segmentation problem, computing meaningful registrations between two *fixed* objects (e.g. whole images, measures, meshes…) has attracted a lot of attention. Typical applications are shape interpolation, data interpretation and using registrations as a basis for a measure of object similarity. Often one requires invariance of the sought-after registration under isometric transformations of either of the two objects. Major approaches include the framework of diffeomorphic matching and metamorphosis [15,39], methods based on physical deformation energies [5,17] and the metric approach to shape matching [7,25]. An extension to shapes that in addition to their geometry are equipped with a 'signal living on the shape' is presented in [9].

These methods provide impressive results at the cost of non-convex functionals and high computational complexity. Naïve online combination with object segmentation is thus not possible. In [32] a shape prior based on object matching has been constructed through convex relaxation of the Gromov-Wasserstein distance [25].

Optimal Transport. Optimal transport is a popular tool in machine learning and image analysis. It provides a meaningful metric on probability measures by 'lifting' a metric from the base space. Thus it is a powerful similarity measure on bag-of-feature representations and other histograms [28]. It is also applied in geometric problems to extract an object registration from the optimal transport plan [16]. However this requires alignment of the objects beforehand. A step towards loosening this constraint is presented in [11] where one optimizes over a suitable class of transformations. The 2-Wasserstein space, induced by optimal transport, exhibits structure akin to a Riemannian manifold [3]. This was exploited in [36] for analysis of spatial variations in observed sets of measures.

1.3 Contribution and Outline

We present a functional for object segmentation with a shape prior. Motivated by the literature on object registration, we propose to base the prior on matching the foreground proposal to a template object. For this we need to be able to jointly optimize over segmentation and registration. Matching is done via optimal transport and based both on geometry and local appearance information. Foreground and template are represented as metric measure spaces [25] which provides ample flexibility. This encompasses a wide range of spatial data structures (pixels, super-pixels, point clouds, sparse interest points,…) and local appearance features (color, patches, filter responses,…). Inspired by [36] the Riemannian structure of the 2-Wasserstein space is used to model geometric transformations, object-typical deformations and changes in appearance in a uniform way. Hence, the resulting approach is invariant under translation and approximately invariant under rotation and scaling.

It has recently been shown that this way of modelling transformations and deformations is equivalent to modelling based on closed contours [31] but *no conversion of shape representation* is required during *inference*. So shape modelling and local appearance matching are performed *directly in the same object representation*, allowing to combine the local appearance matching of indicator functions with the manifold based shape modelling on contours. Also, explicitly using the conversion during *learning* greatly simplifies statistical analysis of the training data and avoids difficulties that arise in [36].

The resulting overall functional is non-convex, but non-convexity is constrained to a low-dimensional variable, making optimization less cumbersome than in typical contour-based approaches or shape matching functionals. Using the linear programming formulation of optimal transport due to Kantorovich, we derive an adaptive convex relaxation and construct a globally optimal branch and bound scheme thereon. Another option is to apply a local alternating optimization scheme. By employing both optimization techniques one after another their respective advantages (no initialization required, speed) can be combined. This allows to construct a 'coarse' object localization method and a subsequent more precise segmentation method as different approximate optimization techniques of the very same functional instead of using two different models. Additionally an efficient graph-cut relaxation is discussed.

Organization. The paper is organized as follows: In Sect. 2 the mathematical background for the paper is introduced. We touch upon the convex variational framework for image segmentation, optimal transport and its differential geometric aspects and the description of shapes via manifolds of (parametrized) contours. The proposed functional is successively developed throughout Sect. 3. We start in Sect. 3.1 with a basic segmentation functional where optimal transport w.r.t. a reference template is used as a shape prior. This functional has obvious limitations (e.g. lack of geometric invariance). An alleviation is proposed in Sect. 3.2 by introducing additional degrees of freedoms that allow transformation of the template set. These transformations can be used to achieve geometric invariance and to model statistical object variation, learned from training data (Sects. 3.3 and 3.4). In Sect. 4 we discuss two different approaches for optimization: locally, based on alternating descending steps and globally by branch and bound with adaptive convex relaxations (Sects. 4.1 and 4.2). A relaxation that replaces optimal transport by graph cuts for reduced computational cost is derived in 4.3. Numerical experiments are presented in Sect. 5 to illustrate the different features of the approach and to compare the two optimization schemes. A brief conclusion is given at the end.

1.4 Notation

For a measure space A we denote by $\text{Meas}(A)$ the set of non-negative and by $\text{Prob}(A)$ the set of probability measures on A. For two measure spaces A, B and a measurable map $f : A \to B$ we write $f_\sharp\mu$ for the push-forward of a measure μ from A to B which is defined by $(f_\sharp\mu)(\sigma) = \mu\big(f^{-1}(\sigma)\big)$ for all measurable $\sigma \subset B$. For $A \subset \mathbb{R}^n$ we denote by $\mathcal{L}_A$ the Lebesgue measure constrained to A and by $|\Omega|$ the Lebesgue volume of a measurable set $\Omega \subset \mathbb{R}^n$. Sometimes, by abuse of notation we use $\mathcal{L}$ to denote the discrete approximation of the Lebesgue measure for discretized domains. For a product space $A \times B$ we denote by $\text{Proj}_A : A \times B \to A$ the canonical projection onto some component. For a differentiable manifold M we write $T_x M$ for the tangent space at footpoint $x \in M$.

2 Mathematical Background

2.1 Convex Variational Image Segmentation

Let $Y \subset \mathbb{R}^2$ be the (continuous) image domain. The goal of object segmentation is the partition of an image into fore- and background. Such a partition can be encoded by an indicator function $u : Y \to \{0, 1\}$ where $u(y) = 1$ encodes that $y \in Y$ is part of the foreground. A typical functional for a variational segmentation approach has the form [22]

$$E(u) = \int_Y s\big(y, u(y)\big)\, dy + R(u)\,. \tag{2.1}$$

The first term is referred to as *data term*, the second as *regularizer*. The data term $s\big(y, u(y)\big)$ describes how well label $u(y)$ matches pixel y, based on local appearance information. The regularizer R introduces prior knowledge to increase robustness to noisy appearance. A common assumption is that boundaries between objects are smooth, a suitable regularizer then is the *total variation*.

To obtain feasible convex problems the constraint that u must be binary is usually relaxed to the interval $[0, 1]$ and the functional (2.1) is suitably extended onto non-binary functions, such that it is convex. In the case of total variation regularization such an extension may be

$$E(u) = \int_Y f(y) \cdot u(y)\, dy + \text{TV}(u) \tag{2.2}$$

where the data term of (2.1) can be equivalently expressed as a linear function in u.

Total variation is a *local* regularizer in the sense that it only depends locally on the (distributional) derivative of its argument. It can thus only account for local noise, i.e. noise that is statistically independent at different points of the image. Although this weakness can be alleviated to some extent by employing non-local total variation [14], the inherent underlying assumption is often not satisfied: faulty observations caused by illumination changes or occlusion clearly have long range correlations. At the same time, in particular for the problem of object segmentation more detailed prior knowledge might be available that is not exploited by local regularizers: the shape of the sought-after object. A non-local regularizer that encourages the foreground region to have a particular shape is called a *shape prior*.

In this article we construct a shape prior by regularization of the foreground region with optimal transport. Hence, we interpret u as the density of a measure ν w.r.t. the Lebesgue measure $\mathcal{L}_Y$ on Y. The feasible set for ν will be:

$$\begin{aligned}\text{SegMeas}(Y, M) = \Big\{\nu \in \text{Meas}(Y) :\ 0 \le \nu \\ \le \mathcal{L}_Y \wedge \nu(Y) = M\Big\}\end{aligned} \tag{2.3}$$

The first constraint ensures that $\nu \in \text{SegMeas}(Y, M)$ has a density which is a relaxed indicator function. The second constraint fixes the overall mass of ν to M. This is necessary to make it comparable by optimal transport.

2.2 Optimal Transport

For two spaces X and Y, two probability measures $\mu \in \text{Prob}(X)$ and $\nu \in \text{Prob}(Y)$ and a cost function $c : X \times Y \to \mathbb{R}$ the optimal transport cost between μ and ν is defined by

$$D(c; \mu, \nu) = \inf_{\pi \in \Pi(\mu,\nu)} \int_{X \times Y} c(x, y)\, d\pi(x, y) \tag{2.4}$$

where

$$\Pi(\mu, \nu) = \Big\{\pi \in \text{Prob}(X \times Y) : \text{Proj}_{X\sharp}\pi = \mu \wedge \text{Proj}_{Y\sharp}\pi = \nu\Big\} \tag{2.5}$$

is referred to as the set of couplings between μ and ν. It is the set of non-negative measures on $X \times Y$ with marginals μ and ν respectively.

For $X = Y = \mathbb{R}^n$ and $c(x, y) = \|x - y\|^2$ one finds that

$$W : \text{Prob}(\mathbb{R}^n)^2 \to \mathbb{R}, \qquad W(\mu, \nu) = (D(c; \mu, \nu))^{1/2} \tag{2.6}$$

is a metric on the space of probability measures on $\mathbb{R}^n$ with finite second order moments, called the 2-Wasserstein space of $\mathbb{R}^n$, here denoted by $\mathcal{W}_2(\mathbb{R}^n)$.

This space exhibits many interesting properties. For example, for two absolutely continuous measures $\mu, \nu \in \mathcal{W}_2(\mathbb{R}^n)$ (2.4) has a unique minimizer $\hat{\pi}$, induced by a map $T : \mathbb{R}^n \to \mathbb{R}^n$, that takes μ onto ν, i.e. $\nu = T_\sharp\mu$ and $\hat{\pi} = (\text{id}, T)_\sharp\mu$ and the measure valued curve

$$[0, 1] \ni \lambda \mapsto \big((1 - \lambda)\,\text{id} + \lambda \cdot T\big)_\sharp \mu \tag{2.7}$$

is a geodesic between μ and ν in $\mathcal{W}_2(\mathbb{R}^n)$. This lead to the observation that the set of absolutely continuous measures in $\mathcal{W}_2(\mathbb{R}^n)$ can informally be viewed as an infinite dimensional Riemannian manifold. The *tangent space* at footpoint μ is represented by gradient fields

$$T_\mu\mathcal{W}_2\left(\mathbb{R}^n\right) = \overline{\left\{\nabla\varphi : \varphi \in C_0^\infty\left(\mathbb{R}^n\right)\right\}}^{L^2(\mu)} \tag{2.8}$$

and the Riemannian inner product for two tangent vectors is given by the L^2 inner product w.r.t. μ:

$$\langle t_1, t_2\rangle_\mu = \int_{\mathbb{R}^n} \langle t_1(x), t_2(x)\rangle_{\mathbb{R}^2}\, d\mu(x) \tag{2.9}$$

Analogous to (2.7) first order variations of a measure μ along a given tangent vector t are described by

$$\lambda \mapsto (\text{id} + \lambda \cdot t)_\sharp\mu. \tag{2.10}$$

The Jacobian determinant of $T_\lambda = \text{id} + \lambda \cdot t$ is

$$\det J_{T_\lambda} = 1 + \lambda \cdot \text{div}\ t + \mathcal{O}(\lambda^2). \tag{2.11}$$

And by the change of variables formula the density of $T_{\lambda\sharp}\mu$ is given by

$$\text{dens}\left(T_{\lambda\sharp}\mu\right)\left(T_\lambda(x)\right) = \text{dens}(\mu)(x) \cdot \left(1 + \lambda \cdot \text{div}\ t(x)\right)^{-1} + \mathcal{O}(\lambda^2). \tag{2.12}$$

Clearly the concept of optimal transport generalizes to non-negative measures of any (finite) mass, as long as the mass of all involved measures is fixed to be identical. An extensive introduction to optimal transport and the structure of Wasserstein spaces is given in [35]. A nice review of the Riemannian viewpoint can be found in [3] and is further investigated in [24] for sufficiently regular measures.

In this paper we will describe the template for our shape prior by a measure μ and model geometric and statistical variations of the shape by tangent vectors $t \in T_\mu\mathcal{W}_2(\mathbb{R}^2)$ and their induced first-order transformations (2.10).

2.3 Contour Manifolds and Shape Measures

The shape of an object can be described by parametrizing its outline contour. Let S^1 denote the unit circle in two dimensions. The set Emb of smooth embeddings of S^1 into $\mathbb{R}^2$ can be treated as an infinite dimensional manifold. A corresponding framework is laid out in [19], a short summary for shape analysis is given in [26]. For various proposed metrics and implementations as shape priors see references in Sect. 1.2. Here we give a very brief summary that aids the understanding of the paper.

The tangent space $T_c\text{Emb}$ at a given curve $c \in \text{Emb}$ is represented by smooth vector fields on S^1, indicating first order deformation:

$$T_c\text{Emb} \simeq C^\infty\left(S^1, \mathbb{R}^2\right) \tag{2.13}$$

This linear structure is a useful basis for analysis of shapes, represented by closed simple contours, and construction of shape priors thereon (see Sect. 1.2).

Let Diff denote the set of smooth automorphisms on S^1. In shape analysis one naturally wants to identify different parametrizations of the same curve. This can be achieved by resorting to the quotient manifold $B = \text{Emb}/\text{Diff}$ of equivalence classes of curves, equivalence $c_1 \sim c_2$ between

$c_1, c_2 \in \text{Emb}$ given if there exists a $\varphi \in \text{Diff}$ such that $c_1 = c_2 \circ \varphi$. We write $[c]$ for the class of all curves equivalent to c.

We summarize:

$$\begin{aligned} \text{Emb} &: \text{smooth embeddings } S^1 \to \mathbb{R}^2 \\ \text{Diff} &: \text{smooth automorphisms on } S^1 \\ B &: \text{quotient Emb/Diff} \end{aligned} \tag{2.14}$$

One finds that for some $a \in T_c\text{Emb}$ the component which is locally tangent to the contour corresponds to a first order change in parametrization of c. 'Actual' changes of the shape can always be represented by scalar functions on S^1 that describe deformations which are locally normal to the contour:

$$H_c\text{Emb} \simeq C^\infty\left(S^1, \mathbb{R}\right) \tag{2.15}$$

where H indicates that this belongs to the *horizontal* bundle on Emb w.r.t. the quotient B. For smooth paths in Emb one can always find an equivalent path such that the tangents lie in H_cEmb. While splitting off reparametrization is very elegant from a mathematical perspective, it remains a computational challenge when handling parametrized curves numerically (see for example [27]).

Alternatively, one can represent a shape by a probability measure with constant density support on the interior of the object. Such measures and their relation to contours have been investigated in [31]. We will here recap the main results. For an embedding $c \in \text{Emb}$ denote by $\Omega(c)$ the region enclosed by the curve and let the map $F : \text{Emb} \to \mathcal{W}_2(\mathbb{R}^2)$ be given by

$$\begin{aligned} &\big(F(c)\big)(A) = |\Omega(c)|^{-1} \cdot |A \cap \Omega(c)| \quad \text{and} \\ &\int \phi\, dF(c) = |\Omega(c)|^{-1} \int_{A\cap\Omega(c)} \phi\, dx \end{aligned} \tag{2.16}$$

for measurable $A \subset \mathbb{R}^2$ and integrable functions ϕ. The set $S = F(\text{Emb})$ of measures is referred to as *shape measures*. If $c_1 \sim c_2$ then obviously $F(c_1) = F(c_2)$, i.e. different parametrizations of the same curve are mapped to the same measure. Thus one can define a map $F_B : B \to \mathcal{W}_2(\mathbb{R}^2)$ by $F_B([c]) = F(c)$ for any representative c of equivalence class $[c]$.

Consider a smooth path $\lambda \mapsto c(\lambda)$ on Emb with tangents $a(\lambda) = \frac{d}{d\lambda} c(\lambda) \in H_{c(\lambda)}\text{Emb}$. The derivative $\frac{d}{d\lambda} F\big(c(\lambda)\big)$ can then be represented by a vector field $t(\lambda) \in T_{F(c(\lambda))}\mathcal{W}_2(\mathbb{R}^2)$ in the distributional sense that for any test function $\phi \in C_0^\infty(\mathbb{R}^2)$ one has

$$\frac{d}{d\lambda} \int \phi\, dF\big(c(\lambda)\big) = \int \langle \nabla\phi, t(\lambda) \rangle_{\mathbb{R}^2}\, dF\big(c(\lambda)\big). \tag{2.17}$$

For a contour c the measure tangent $t \in T_{F(c)}\mathcal{W}_2(\mathbb{R}^2)$ at $F(c)$ corresponding to a contour tangent $a \in H_c\text{Emb}$ at contour c in the sense of (2.17), one has on $\Omega(c)$ that $t = \nabla u$ where u solves the Neumann problem

$$\Delta u = C \quad \text{in} \quad \Omega(c), \quad \frac{\partial u}{\partial n} = a \circ c^{-1} \quad \text{on} \quad \partial\Omega(c) \tag{2.18a}$$

with $\frac{\partial}{\partial n}$ denoting the derivative in outward normal direction of the contour and

$$C = |\Omega(c)|^{-1} \int_{\partial\Omega(c)} a \circ c^{-1}\, ds \tag{2.18b}$$

is the normalized total flow of a through the surface $\partial\Omega(c)$. This maps a to a uniquely determined t. We denote this map by f_c (depending on the basis contour c) and write $t = f_c(a)$.

Note that $t = f_c(a)$ has constant divergence on $\Omega(c) = \operatorname{spt} F(c)$. Hence by virtue of (2.12) one finds to first order of λ that $\mu(\lambda) = (\text{id} + \lambda \cdot t)_\sharp F(c)$ has constant density on its support and is therefore itself a shape measure.

So vector fields generated as $t = f_c(a)$ can said to be tangent to the set S in $\mathcal{W}_2(\mathbb{R}^2)$ and the former can informally be regarded as a submanifold of the latter. When equipped with the proper topology it becomes a manifold in the sense of [19] which is diffeomorphic to B.

This means that describing shapes via shape measures and appropriate tangent vectors thereon is mathematically equivalent to describing shapes by contours modulo parametrization and deformations. Thus we can *construct shape priors for regularization with optimal transport*, based on measures, *without any representation conversion during inference and without having to handle parametrization ambiguities numerically*.

3 Regularization with Optimal Transport

3.1 Setup and Basic Functional

Let $Y \subset \mathbb{R}^2$ describe the image domain in which we want to locate and match the sought-after object. As discussed in Sect. 2.1 we will describe the object location by a relaxed indicator function $u : Y \to [0, 1]$. Since we want to use optimal transport for regularization, u will be interpreted as density of a measure ν. The feasible set for ν is given by $\text{SegMeas}(Y, M)$ as defined in (2.3) where M is the total mass of the reference measure which we use for regularization.

Note that this is conceptually different from matching approaches where a certain local image feature (usually intensity or gray-level) is directly converted into a density. The limitations of this are discussed in [9] in the context of 'colored currents'. In brief, one problem is, for example, that only one dimensional features can be described. Another

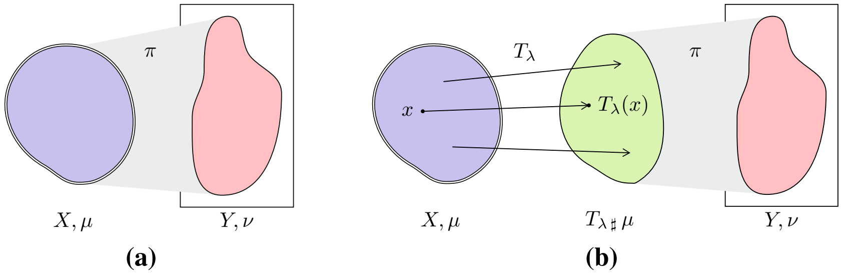


Fig. 1 Illustration of functionals $E(\nu)$, Eq. (3.3), and $E(\lambda, \nu)$, Eq. (3.5): **a** The segmentation in Y is described by measure ν which is regularized by the Wasserstein distance to a template measure μ, living on X. This simple approach introduces strong bias, depending on the relative location of X and Y, and lacks the ability to explicitly model typical object deformations. **b** In the enhanced functional the template measure μ is deformed by the map T_λ, resulting in the push-forward $T_{\lambda\,\sharp}\,\mu$. The segmentation ν is then regularized by its Wasserstein distance to $T_{\lambda\,\sharp}\,\mu$. The corresponding optimal coupling π gives a registration between the foreground part of the image and the deformed template (Color figure online)

is, that, by converting features to density, different, a priori equally important image regions, are assigned different densities and thus have a different influence on the optimizer.

We use the measure to indicate the *location* of the sought-after object. Local image data is handled in a *different* fashion: for this we introduce a suitable *feature space* $\mathcal{F}$. Depending on the image this may be the corresponding color space. It may however also be a more elaborate space spanned by small image patches or local filter responses. We then assume that any point $y \in Y$ is equipped with some $f_y \in \mathcal{F}$ which we refer to as the *observed feature*. We can thus consider every pixel to be a point in the enhanced space $Y \times \mathcal{F}$ with coordinates (y, f_y).

For regularization with optimal transport we need to provide a prototype, referred to as *template*. Let X be a set whose geometry will model the shape of the object of interest. It will be equipped with a measure μ which should usually be the Lebesgue measure on X, having density 1 everywhere, to indicate that 'all of X is part of the object'. The constant M specifying the total mass for feasible segmentations ν will be the mass of μ:

$$M = \mu(X) \tag{3.1}$$

Additionally, we describe the *appearance* of the template by associating to all elements $x \in X$ corresponding $f_x \in \mathcal{F}$, the *expected features*.

We assume that both the template X and the image domain Y are embedded into $\mathbb{R}^2$. The squared Euclidean distance $\|x - y\|^2$ for $x \in X$ and $y \in Y$ then provides a geometric matching cost for points:

$$c_{\text{geo}}(x, y) = \|x - y\|^2 \tag{3.2}$$

Moreover, we pick some function $c_{\mathcal{F}} : \mathcal{F} \times \mathcal{F} \to \mathbb{R}$ which models the matching cost on the feature space. Possible choices for $c_{\mathcal{F}}$ are for example a (squared) metric, or a Bayesian log-likelihood for observing a noisy feature f_y when expecting feature f_x.

Combining this, we can construct a functional for rating the plausibility of a segmentation proposal $\nu \in \text{SegMeas}(Y, M)$:

$$\begin{aligned} E(\nu) = \frac{1}{2} \inf_{\pi \in \Pi(\mu,\nu)} \int_{X \times Y} \Big(& c_{\text{geo}}(x, y) \\ & + c_{\mathcal{F}}(f_x, f_y) \Big)\, d\pi(x, y) + G(\nu) \end{aligned} \tag{3.3}$$

The first term is the minimal matching cost between the segmentation region and the template via optimal transport with a cost function that combines the geometry and appearance. The second term can contain other typical components of a segmentation functional, for example a local boundary regularizer (cf. Sect. 2.1). The functional is illustrated in Fig. 1a.

Remark 3.1 (Generality of functional). Although we describe here a continuous setup, numerically functional (3.3) can be applied to a wide range of different data structures. X and Y can be open sets in $\mathbb{R}^2$, describing continuous templates and images. Then μ would be, as indicated, the Lebesgue measure on X and $\mathcal{L}_Y$ in (2.3) would be the Lebesgue measure on Y. Alternatively X and Y could be discrete sets of pixels in $\mathbb{R}^2$ or point clouds in $\mathbb{R}^n$, then μ and $\mathcal{L}_Y$ should be chosen to be the respective uniform counting measures on X and Y. If X and Y represent an over-segmentation of some data (i.e. super-pixels or voxels), μ and $\mathcal{L}_Y$ would be weighted counting measures, the weights representing the area/volume of each cell.

Remark 3.2 (Metric structure of $\mathcal{W}_2(\mathbb{R}^2)$). Adding the term $c_{\mathcal{F}}$ to the optimal transport cost breaks the geometric structure of $\mathcal{W}_2(\mathbb{R}^2)$, therefore some readers may be hesitant about this step. However the measure ν is an unknown variable in the approach. Therefore numerical solvers that rely on the $\mathcal{W}_2(\mathbb{R}^2)$-structure cannot be applied directly, even without the $c_{\mathcal{F}}$ term. Instead we use discrete solvers in this paper, which can simultaneously optimize for ν and π. So $c_{\mathcal{F}}$ does not add any computational complexity whereas we gain significantly more modelling flexibility. Additionally, when one chooses $c_{\mathcal{F}}$ to be a squared metric on $\mathcal{F}$, then one is working on $\mathcal{W}_2(\mathbb{R}^2 \times \mathcal{F})$, which also exhibits a metric structure.

Limitations of the Basic Functional. Functional (3.3) has three major shortcomings for the application of object segmentation and shape matching, related to the choice of the embedding $X \to \mathbb{R}^2$:

(i) The location and orientation of the sought-after object are often unknown beforehand. Hence, a segmentation method should be invariant under Euclidean isometries, which is clearly violated by picking an arbitrary embedding $X \to \mathbb{R}^2$. If μ and ν were fixed measures in $\text{Meas}(\mathbb{R}^2)$ with equal mass, then the optimal coupling for $W(\mu, \nu)$ would be invariant under translation (up to an adjustment of the coordinates according to the translation, of course). However, since in this application ν is not fixed this quasi-invariance cannot be exploited. Also, there is no similar invariance w.r.t. rotation.

(ii) Any non-isometric deformation between template foreground and the object will be uniformly penalized by the geometric part of the corresponding optimal transport cost. No information on more or less common deformations (learned from a set of training samples) can be encoded.

(iii) Since the mass M of μ, related to the size of the template X, equals the mass of ν, this determines the size of the foreground object in Y. Hence, the presented functionals imply that one must know the scale of the sought-after object beforehand. This is not possible in all applications.

In the next sections we will discuss how to overcome these obstacles. By making the embedding $X \to \mathbb{R}^2$ flexible, the resulting functionals become fit for (almost) isometry invariance, can handle prior information on more or less common non-isometric deformations and can dynamically adjust the object scale.

3.2 Wasserstein Modes

To overcome the limitations listed in Sect. 3.1 we will allow X to move and be deformed within $\mathbb{R}^2$. We choose the following family of embeddings:

$$T_\lambda : X \to \mathbb{R}^2, \qquad T_\lambda(x) = x + \sum_{i=1}^{n} \lambda_i \cdot t_i(x),$$
$$t_i \in T_\mu \mathcal{W}_2\left(\mathbb{R}^2\right) \tag{3.4}$$

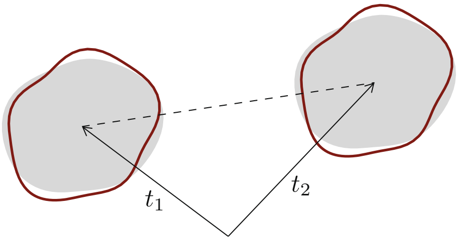


Fig. 2 Explicit transformation variables and non-convexity. *Gray shading* indicates 'foreground features'. Placing the template (*red contour*) at t_1 or t_2 yields equally good hypotheses. Were the prior functional convex in the translation variable, any point along the line $(1-\alpha) \cdot t_1 + \alpha \cdot t_2$ for $\alpha \in [0, 1]$ would yield an at least equally good proposal, which is clearly unreasonable (Color figure online)

The transformation is parametrized by the coefficients $\lambda \in \mathbb{R}^n$. This linear decomposition will allow enough flexibility for modelling while keeping the resulting functionals amenable. We refer to the basis maps $\{t_i\}_{i=1}^n$ as *modes*. Including the coefficients λ as degrees of freedom into (3.3) yields:

$$E(\lambda, \nu) = \frac{1}{2} \inf_{\pi \in \Pi(\mu,\nu)} \int_{X \times Y} \Big(c_{\text{geo}}\big(T_\lambda(x), y\big) + c_{\mathcal{F}}(f_x, f_y) \Big)\, d\pi(x, y) + F(\lambda) + G(\nu) \tag{3.5}$$

The function F can be used to introduce statistical knowledge on the distribution of the coefficients λ. The enhanced functional is illustrated in Fig. 1b.

Functional (3.5) is generally non-convex. For fixed λ it is convex in ν. For fixed ν and a fixed coupling π in the optimal transport term it is convex in λ if transformations are of the form (3.4) and F is convex. Joint non-convexity does not come as a surprise. It is in fact easy to see that a meaningful isometry invariant segmentation functional with explicitly modelled transformations is bound to be non-convex (Fig. 2).

Remark 3.3 (Eliminating ν). For optimization of (3.5) assume we first eliminate the high-dimensional variable ν through minimization (which is a convex problem). One is then left with:

$$E_1(\lambda) = \inf_{\nu \in \text{SegMeas}(Y,M)} E(\lambda, \nu) \tag{3.6}$$

This is in general non-convex, but the dimensionality of λ is typically very low (of the order of 10). We can thus still hope to find globally optimal solutions by means of non-convex optimization. We will present a corresponding branch and bound scheme in Sect. 4.2.

Remark 3.4 (Modelling transformations in feature space) When the feature space $\mathcal{F}$ has an appropriate linear structure a natural generalization of (3.4) is to not only model geometric transformations of X but also of the expected features f_x. In analogy to (3.4) consider

$$\hat{T}_\lambda : X \to \mathbb{R}^2 \times \mathcal{F}, \qquad \hat{T}_\lambda(x) = (x, f_x) + \sum_{i=1}^{n} \lambda_i \cdot \hat{t}_i(x) \tag{3.7}$$

where $\hat{T}_0(x) = (x, f_x)$ returns the original position and expected feature of a point. The modes $\hat{t}_i : X \to \mathbb{R}^2 \times \mathcal{F}$ can then be used to *alter both the geometry of* X *as well as its appearance*.

This will be useful when the appearance of the object is known to be subject to variations or when a feature is affected by geometric transformations: for example the expected response to an oriented local filter will need to be changed when the object is rotated. The corresponding *generalized functional* is

$$\begin{aligned} E_\mathcal{F}(\lambda, \nu) = \frac{1}{2} \inf_{\pi \in \Pi(\mu,\nu)} \int_{X \times Y} \hat{c}\big(\hat{T}_\lambda(x), (y, f_y)\big)\, d\pi(x, y) \\ + F(\lambda) + G(\nu) \end{aligned} \tag{3.8}$$

with

$$\begin{aligned} &\hat{c} : (\mathbb{R}^2 \times \mathcal{F})^2 \to \mathbb{R}, \qquad \hat{c}\big((x', f'_x), (y, f_y)\big) \\ &\quad = c_{\text{geo}}(x', y') + c_\mathcal{F}(f'_x, f_y). \end{aligned} \tag{3.9}$$

We will further study this generalization in Sect. 5. Meanwhile, for the sake of simplicity we constrain ourselves to purely geometric modes.

In this paper we assume that $X = \Omega(c)$ for some $c \in \text{Emb}$ (cf. Sect. 2.3). As pointed out, for a meaningful template μ should be the Lebesgue measure on X with constant density 1, so μ is a (rescaled) shape measure. The modes $\{t_i\}_i$ span a subspace of $T_\mu \mathcal{W}_2(\mathbb{R}^2)$ in which λ parametrizes a first-order deformation. We will choose $t_i \in T_\mu S$, i.e. tangents to the manifold of shape measures. This is equivalent to the tangent space approximation of the contour manifold B modulo parametrizations. We need to take into account how transforming X through T_λ alters μ. We discussed earlier that according to (2.10) the density of $T_{\lambda\sharp}\mu$ remains constant to first order. Modes with non-zero divergence will lead to a density which is not 1. Hence, $T_{\lambda\sharp}\, \mu$ must be rescaled accordingly, which will change its total mass and thus influence the corresponding feasible set SegMeas(Y, M) for ν. This will require some additional care during optimization. All constant-divergence modes can be decomposed into zero-divergence modes plus an additional 'scale mode' (see Sect. 3.3). We will thus see to it that all but one mode will have zero divergence and handle the scale mode with particular care (Sect. 4).

3.3 Geometric Invariance

The framework provided by transformations (3.4) and functional (3.5) allows to introduce geometric invariance into the segmentation/matching approach. In this section we will consider translations, (approximate) rotations and scale transformations. Scale transformations will play a special role as they change the mass of the template.

The transformations will be modelled with the generators of the corresponding (local) Lie group acting on $\mathbb{R}^2$. Likewise invariance w.r.t. transformation Lie groups could be introduced into matching functionals on other manifolds.

Translation and Rotation. If one chooses modes

$$t_{\text{t}1}(x) = (1, 0)^\top, \qquad t_{\text{t}2}(x) = (0, 1)^\top \tag{3.10}$$

the corresponding coefficients $\lambda_{\text{t}1}, \lambda_{\text{t}2}$ parametrize translations of the template. Further, let $R(\phi)$ be the 2-dimensional rotation matrix by angle ϕ. Then the mode

$$t_\text{r}(x) = \frac{d}{d\phi} R(\phi)\bigg|_{\phi=0} x = (-x_2, x_1)^\top \tag{3.11}$$

will approximately rotate the template.[1] This first order expansion works satisfactory for angles up to about ±30°. We will consider larger rotations in the experiments, Sect. 5.

Note that $t_{\text{t}1}$, $t_{\text{t}2}$ and t_r have zero divergence. Hence, to first order the implied transformations do not alter the density of μ. For explicit invariance under translations and rotations the modelling function F in (3.5) should be constant w.r.t. the coefficients $\lambda_{\text{t}1}, \lambda_{\text{t}2}$ and λ_r.

Scale. The size of X and μ determines the size of the object within the image. In many applications the scale is not known beforehand, thus dynamical resizing of the template during the search is desirable. With slight extensions the framework of transformations can be employed to introduce as a scale-mode into the approach. Let

$$t_\text{s}(x) = x. \tag{3.12}$$

By the change of variable formula (cf. (2.11, 2.12)) the density of $T_{\lambda\sharp}\, \mu$ is given by

$$\text{dens}\left(T_{\lambda\sharp}\, \mu\right)\left(T_\lambda(x)\right) = \text{dens}(\mu)(x) \cdot \left(\det J_{T_\lambda}(x)\right)^{-1}. \tag{3.13}$$

[1] Note that t_r is not a gradient field and thus $\notin T_\mu \mathcal{W}_2(\mathbb{R}^2)$. One could find a corresponding gradient version by lifting the rotation field from the contour to the interior, Sect. 2.3. However the functional is also meaningful with this non-gradient mode and its effect on the template is more intuitive.

By plugging in the scale mode t_s and ignoring other modes, which due to zero divergence do not contribute to first order, we find in 2 dimensions:

$$= (1 + \lambda_\mathrm{s})^{-2} \tag{3.14}$$

Thus, introducing a scale mode into (3.5) yields

$$E_\mathrm{s}(\lambda, \nu) = \frac{1}{2\,(1+\lambda_\mathrm{s})^2} \inf_{\pi \in \Pi\left((1+\lambda_\mathrm{s})^2 \cdot \mu, \nu\right)} \int_{X \times Y} \Big(c_\mathrm{geo}\big(T_\lambda(x), y\big) + c_\mathcal{F}(f_x, f_y) \Big)\, d\pi(x, y) + F(\lambda) + G(\nu) \tag{3.15}$$

where we have scaled μ by the appropriate factor in the feasible set for π and we have normalized the first term by a factor of $(1+\lambda_\mathrm{s})^{-2}$ to make the term scale invariant. Depending on whether scale invariance is desired the terms $F(\lambda)$ and $G(\nu)$ may need to be rescaled appropriately, too. The feasible set for ν in E_s is $\mathrm{SegMeas}\big(Y, (1+\lambda_\mathrm{s})^2 \cdot M\big)$.

While the modes for translation and rotation leave the area of the template unaltered, statistical deformation modes that we learn from sample data will in general have non-zero divergence. Handling changes in mass will require some extra care during optimization. Therefore we will decompose such modes into a divergence-free part and a contribution of the scale-component.

3.4 Statistical Variation

One of the limitations of (3.3) discussed in Sect. 3.1 is that non-isometric variations of the template object are uniformly penalized by the geometric component of the corresponding optimal transport cost. However, not all deformations with the same optimal transport cost are equally likely. It may be necessary to reweigh the distance to more accurately model common and less common deformations.

For contour based shape priors a model of statistical object variations is typically learned from samples in a tangent space approximation of the contour manifold. In [36] the tangent space approximation to the Wasserstein space $\mathcal{W}_2$ was used to analyze typical deformations in a dataset of densities. But mimicking the learning procedure on the contour manifold with optimal transport involves some unsolved problems.

(i) The first problem is to find an appropriate footpoint for the tangent space approximation, i.e. a point by the associated tangent space of which we want to approximate the manifold to first order. One should pick a point which is close to all training samples. Typically one chooses a suitable mean, in a more general metric setting the natural generalization is the Karcher mean. Computation of the barycenter on $\mathcal{W}_2$ is a non-trivial problem [2], which has recently been made more accessible through Entropy smoothing [13]. However it becomes more involved when one wants to take geometric invariances into account and impose the constraint of constant density on the support. In [36] the L^2-mean of the density functions was picked as footpoint after aligning the centers of mass and the principal axes of the samples. Though this is not necessarily an ideal choice (the L^2-mean of the densities can be very far from some of the samples) it seems to work for smooth densities with limited variations. It will not extend to the binary densities that we consider in this paper since their L^2-mean need not be binary. In [33] the problem was tentatively solved by manually picking a 'typical' sample from the training set as the footpoint.

(ii) The second problem is how one maps the samples into the tangent space of the footpoint. A natural choice is the logarithmic map, or some approximation thereof. Recall from Sect. 2.2 that tangent vectors on the manifold of measures are curl-free vector fields and that the logarithmic map is basically obtained by taking the relative transport map. There are some issues with the application to object segmentation: The vector fields computed by the logarithmic map need not have constant divergence, although fluctuations are typically small enough to be ignored for practical purposes. A second issue is that the vector fields are in general not smooth between measures with non-smooth densities, as in our case. This leads to unreasonable interpolations and unwanted artifacts during statistical analysis of the vector fields representing the sample set.

In this paper we circumvent both problems by employing the diffeomorphism between the manifold of contours and the manifold of shape measures (see Sect. 2.3). This allows us to outsource the shape learning problem to the contour representation where established methods for finding a good mean and tangent vectors are available (for example [27]).

Concretely we used the contour metric and the corresponding approximate algorithmic framework based on gradient descent and dynamic programming presented in [27] for computing the Karcher mean of a set of training shapes and for mapping the training-samples onto the tangent space at the mean via the logarithmic map. We then performed a principal component analysis w.r.t. the Riemannian inner product to extract the dominating modes of shape variation within the training set, together with their observed standard deviation $\{(t_i, \sigma_i)\}$. The results we obtained were stable under choosing different initializations. Learning of the class 'starfish' is illustrated in Fig. 3. The standard deviations σ_i were then used to define $F(\lambda)$ to model a Gaussian distribution on the statistical mode parameters:

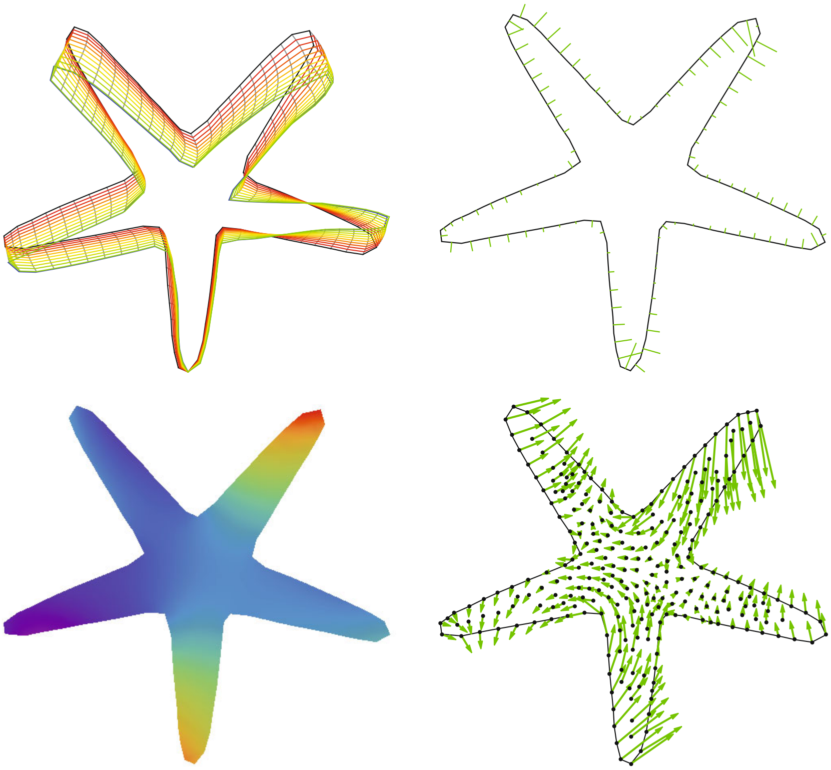


Fig. 3 Learning of contours. *Top left:* geodesic from shape mean to a training sample. *Top right:* Normal contour deformation of first principal component of training samples. *Bottom left:* Potential function u for lifting the deformation to the full region (see (2.18)). *Bottom right:* Gradient field which gives deformation mode for whole template region (Color figure online)

$$F(\lambda) = \frac{\gamma}{2} \sum_{i=1}^{n_{\text{stat}}} \left(\frac{\lambda_i}{\sigma_i}\right)^2 \tag{3.16}$$

where γ is a parameter determining the weight of F w.r.t. the other functional components.

3.5 Background Modelling

The previous sections describe how to model the sought-after object via a template, i.e. they focus on the image foreground. Let us now briefly comment on the background.

Sometimes information on the expected appearance of the background is available. This can be incorporated by a linear contribution to G (3.5):

$$G(\nu) = \int_Y g(y)\, d\nu(y) \tag{3.17}$$

where a positive (negative) coefficient $g(y)$ indicates that a given point is likely to be part of the background (foreground) (cf. Sect. 2.1). Such linear terms can be absorbed into the optimal transport term:

$$\int_Y g(y)\, d\nu(y) = \int_{X\times Y} g(y)\, d\pi(x,y) \tag{3.18}$$

That is, the background appearance model leads to an effective shift of the foreground assignment costs: $c(x,y) \to c(x,y) + g(y)$.

In other situations it may be desirable to impose that the region directly around the foreground object does not look like foreground itself. An example for such a situation and the corresponding solution are discussed with numerical examples in Sect. 5, see Fig. 5.

4 Optimization

4.1 Alternating Optimization

Functional (3.5) is generally non-convex. It is convex in ν for fixed λ and it is convex in λ under suitable conditions (see Sect. 3.2). Based on this, an alternating optimization scheme is conceivable for divergence-free modes. This has also been proposed in [11, Sect. 3.2.1]. We require the following reformulation of (3.6):

Remark 4.1 (Coupling reformulation). Computing (3.6) involves a nested optimization problem over $\nu \in \text{SegMeas}(Y, M)$ and then $\pi \in \Pi(\mu,\nu)$. Given a coupling $\pi \in \Pi(\mu,\nu)$ the marginal ν can be reconstructed via projection: $\nu = \text{Proj}_{Y\sharp}\pi$. This allows to reformulate the optimization of (3.6) directly in terms of couplings. Let

$$\hat{E}(\lambda, \pi) = \frac{1}{2} \int_{X \times Y} \Big(c_{\text{geo}}\big(T_\lambda(x), y\big) + c_{\mathcal{F}}(f_x, f_y) \Big)\, d\pi(x, y) + F(\lambda) + G(\text{Proj}_{Y\sharp}\pi) \tag{4.1}$$

and let the feasible set for π in $\hat{E}$ be

$$\begin{aligned} \text{SegCoupl}(Y, \mu) &= \bigcup_{\nu \in \text{SegMeas}(Y,M)} \Pi(\mu, \nu) \\ &= \Big\{ \pi \in \text{Meas}(X \times Y) : \text{Proj}_{X\sharp}\pi \\ &= \mu \wedge \text{Proj}_{Y\sharp}\pi \leq \mathcal{L}_Y \Big\}. \end{aligned} \tag{4.2}$$

Then for fixed λ one has by construction

$$\inf_{\nu \in \text{SegMeas}(Y,M)} E(\lambda, \nu) = \inf_{\pi \in \text{SegCoupl}(Y,\mu)} \hat{E}(\lambda, \pi) \tag{4.3}$$

and for any optimizer π^* of $\hat{E}$ the marginal $\text{Proj}_{Y\sharp}\pi^*$ is an optimizer of E.

Functional $\hat{E}(\lambda, \pi)$ is separately convex in λ and π for transformations of the form (3.4) and convex F. For some initial λ^1 consider the following sequence for $k = 1, 2, \ldots$:

$$\pi^k \in \text{argmin}_{\pi \in \text{SegCoupl}(Y,\mu)} \hat{E}(\lambda^k, \pi) \tag{4.4a}$$

$$\lambda^{k+1} \in \text{argmin}_{\lambda \in \mathbb{R}^n} \hat{E}(\lambda, \pi^k) \tag{4.4b}$$

Proposition 4.2 *The sequence of energies $\hat{E}(\lambda^1, \pi^1) \to \hat{E}(\lambda^2, \pi^1) \to \hat{E}(\lambda^2, \pi^2) \to \ldots$ is non-increasing and converges.*

Proof Since λ^k is feasible when determining λ^{k+1}, one has $\hat{E}(\lambda^{k+1}, \pi^k) \leq \hat{E}(\lambda^k, \pi^k)$. Likewise π^k is a feasible point for computing π^{k+1} so $\hat{E}(\lambda^{k+1}, \pi^{k+1}) \leq \hat{E}(\lambda^{k+1}, \pi^k)$. Hence, the sequence of energies is non-increasing. As $\hat{E}$ is bounded from below, the sequence of energies must converge. □

Unfortunately this cannot be extended to modes with non-zero divergence, as changing λ_s changes the feasible set $\text{SegCoupl}\big(Y, (1 + \lambda_s)^2 \cdot \mu\big)$ for π. Thus π^k need not be feasible for the problem that determines π^{k+1} and the sequence of energies created may be increasing. We will provide a workaround for this in the next section (Remark 4.7).

The alternating scheme (4.4) is fast and tends to converge after few iterations. But obviously it need not converge to a global optimum and the result depends on the initialization λ^1. Therefore, similar to contour based segmentation functionals it must be applied with care. In practice application to 'large' transformations, e.g. translations and rotations, works only if a good initial guess is available (see Fig. 9). On the other hand it achieves decent results on smaller transformations, as most statistically learned deformations are.

4.2 Globally Optimal Branch and Bound

For handling large displacement transformations, one needs a global optimization scheme. As discussed in Remark 3.3, for fixed λ we can eliminate ν by a separate convex optimization. One obtains (3.6):

$$\begin{aligned} E_1(\lambda) &= \inf_{\nu \in \text{SegMeas}(Y,M)} E(\lambda, \nu) \\ &= \inf_{\nu \in \text{SegMeas}(Y,M)} \frac{1}{2} \inf_{\pi \in \Pi(\mu,\nu)} \int_{X \times Y} \Big(c_{\text{geo}}\big(T_\lambda(x), y\big) \\ &\quad + c_{\mathcal{F}}(f_x, f_y) \Big)\, d\pi(x, y) + F(\lambda) + G(\nu) \end{aligned} \tag{4.5}$$

This function is in general non-convex but low dimensional. We thus strive for a non-convex global optimization scheme.

Given Remark 4.1 $E_1(\lambda)$ can be written as

$$E_1(\lambda) = \inf_{\pi \in \text{SegCoupl}(Y,\mu)} \frac{1}{2} \int_{X \times Y} \Big(c_{\text{geo}}\big(T_\lambda(x), y\big) + c_{\mathcal{F}}(f_x, f_y) \Big)\, d\pi(x, y) + F(\lambda) + G(\text{Proj}_{Y\sharp}\pi). \tag{4.6}$$

If G is zero, then by inserting suitable dummy nodes, computing $E_1(\lambda)$ can be written as an optimal transport problem for which efficient solvers are available.

In this section we will consider a hierarchical *branch and bound* approach. We will compute lower bounds for E_1 on whole intervals of λ-configurations for successively refined intervals. Let $\Lambda \subset \mathbb{R}^n$ be a set of λ-values. We assume for now that all modes have zero divergence. For such subsets define

$$E_2(\Lambda) = \inf_{\pi \in \text{SegCoupl}(Y,\mu)} \frac{1}{2} \int_{X \times Y} \bigg(\Big(\inf_{\lambda \in \Lambda} c_{\text{geo}}\big(T_\lambda(x), y\big) \Big) + c_{\mathcal{F}}(f_x, f_y) \bigg)\, d\pi(x, y) + \inf_{\lambda \in \Lambda} F(\lambda) + G(\text{Proj}_{Y\sharp}\pi) \tag{4.7}$$

where we have again merged the nested optimizations as above. All occurrences of λ are optimized separately and independently over Λ. By introducing a nested sequence of feasible sets

$$\Lambda_1 \supset \Lambda_2 \supset \cdots \supset \Lambda_n \tag{4.8}$$

we obtain an adaptive convex relaxation of $E_1(\lambda)$ over Λ. The relaxation becomes tighter as the set becomes smaller. For application in a branch and bound scheme the following properties are required:

Proposition 4.3 ([33, Prop. 1]). *The functional E_2 has the following properties:*

(i) $E_2(\Lambda) \leq E_1(\lambda)\,\forall\,\lambda \in \Lambda$,
(ii) $\lim_{\Lambda\to\{\lambda_0\}} E_2(\Lambda) = E_1(\lambda_0)$,
(iii) $\Lambda_1 \subset \Lambda_2 \Rightarrow E_2(\Lambda_1) \geq E_2(\Lambda_2)$.

Proof Property (i): For any $\lambda \in \Lambda$ obviously

$$\inf_{\lambda'\in\Lambda} c_{\text{geo}}\big(T_{\lambda'}(x), y\big) \leq c_{\text{geo}}\big(T_\lambda(x), y\big) \qquad \text{and}$$
$$\inf_{\lambda'\in\Lambda} F(\lambda') \leq F(\lambda). \tag{4.9}$$

So for any fixed $\pi \in \text{SegCoupl}(Y,\mu)$ (overriding the minimization in (4.6, 4.7)) have $E_2(\Lambda) \leq E_1(\lambda)$. Consequently this inequality will also hold after minimization w.r.t. π.

For the limit property (ii) note that the functions $c_{\text{geo}}\big(T_\lambda(x), y\big)$ and $F(\lambda)$ are continuous functions of λ. Hence, when $\Lambda \to \{\lambda_0\}$ all involved minimizations will converge towards the respective function values at λ_0 and E_2 converges as desired.

For the hierarchical bound property (iii) note that for fixed π in (4.7) minimization over the larger set Λ_2 will never yield the larger result for all occurrences of λ. This relation will then also hold after minimization. □

With the aid of E_2 one can then construct a branch and bound scheme for optimization of E_1. Let

$$L = \{(\Lambda_i, b_i)\}_{i\in\{1,\dots,k\}} \tag{4.10}$$

be a finite list of λ-parameter sets Λ_i and lower bounds b_i on E_1 on these respective sets. For such a list consider the following refinement procedure:
`refine`**(L):**

(1) Find the element $(\Lambda_{i^*}, b_{i^*}) \in L$ with the smallest lower bound b_{i^*}.
(2) Let $\texttt{subdiv}(\Lambda_{i^*}) = \{\Lambda_{i^*,j}\}_j$ be a subdivision of the set Λ_{i^*} into smaller sets.
(3) Compute $b_{i^*,j} = E_2(\Lambda_{i^*,j})$ for all $\Lambda_{i^*,j} \in \texttt{subdiv}(\Lambda_{i^*})$.
(4) Remove (Λ_{i^*}, b_{i^*}) from L and add $\{(\Lambda_{i^*,j}, b_{i^*,j})\}_j$ for $\Lambda_{i^*,j} \in \texttt{subdiv}(\Lambda_{i^*})$.

This allows the following statement:

Proposition 4.4 ([33, Prop. 2]). *Let L be a list of finite length. Let the subdivision in* `refine` *be such that any set will be split into a finite number of smaller sets, and that any two distinct points will eventually be separated by successive subdivision. Set* $\texttt{subdiv}(\{\lambda_0\}) = \{\{\lambda_0\}\}$. *Then repeated application of* `refine` *to the list L will generate an adaptive piecewise constant underestimator of E_1 throughout the union of the sets Λ appearing in L. The sequence of smallest lower bounds will converge to the global minimum of E_1.*

Proof Obviously the sequence of smallest lower bounds is non-decreasing and never greater than the minimum of E_1 throughout the considered region (see Proposition 4.3 (iii) and (i)). So it must converge to a value which is at most this minimum. Assume that $\{\Lambda_i\}_i$ is a sequence with $\Lambda_{i+1} \in \texttt{subdiv}(\Lambda_i)$ such that $E_2(\Lambda_i)$ is a subsequence of the smallest lowest bounds of L (there must be such a sequence since L is finite). Since `subdiv` will eventually separate any two distinct points, this sequence must converge to a singleton $\{\lambda_0\}$ and the corresponding subsequence of smallest lowest bounds converges to $E_2(\{\lambda_0\}) = E_1(\lambda_0)$. Since the sequence of smallest lowest bounds converges, and the limit is at most the minimum of E_1, $E_1(\lambda_0)$ must be the minimum. □

When the global optimum is unique, one can see that there also is a subsequence of λ-sets, converging to the global optimum.

In practice we start with a coarse grid of hypercubes covering the space of reasonable λ-parameters (e.g. translation throughout the image, rotation within bounds where the approximation is valid and the deformation-coefficients in ranges according to the statistical model) and the respective E_2-bounds. Any hypercube with the smallest bound will then be subdivided into equally sized smaller hypercubes, leading to an adaptive 2^n-tree cover on the considered parameter range.

The refinement is stopped, when the interval with the lowest bound has edge lengths that correspond to an uncertainty in $T_\lambda(x)$ which is in the range of the discretization of X and Y. Further refinement would only reveal structure determined by rasterization effects.

Remark 4.5 (Combining hierarchical and alternating optimization). The optimum of E_1 w.r.t. modes that have large displacements (such as translation and rotation) tends to be rather distinct, i.e. there is a small, steep basin around the optimal position. The hierarchical optimization scheme then works rather efficiently.

On the other hand, modes that model smaller, local displacements (e.g. those learned from training samples), often have broad, shallow basins around the optimal value. The branch and bound scheme can then take longer to converge.

Therefore it suggests itself to combine the two optimization schemes: the hierarchical approach is used to determine a good initial guess for translation, rotation and a coarse estimate for the smaller modes. For this the alternating scheme is not applicable due to the non-convexity. But once the broad basin around the global optimum is located, the branch and bound scheme may become inefficient. Conversely, using the estimate of the hierarchical scheme as initialization, we can then expect that the alternating method will give reasonable results.

Scale Mode. In the presence of a scale mode one can define $E_{\text{s},1}$ and $E_{\text{s},2}$ equivalent to E_1 and E_2 with slight adaptations.

$$\begin{aligned}E_{\text{s},1}(\lambda) &= \inf_{\nu \in \text{SegMeas}\left(Y,(1+\lambda_\text{s})^2 \cdot M\right)} E_\text{s}(\lambda,\nu) \\ &= \inf_{\pi \in \text{SegCoupl}\left(Y,(1+\lambda_\text{s})^2 \cdot \mu\right)} \frac{1}{2\,(1+\lambda_\text{s})^2} \\ &\quad \int_{X \times Y} \left(c_\text{geo}\left(T_\lambda(x), y\right) + c_\mathcal{F}(f_x, f_y)\right) d\pi(x,y) \\ &\quad + F(\lambda) + G(\text{Proj}_{Y\sharp}\pi) \qquad (4.11)\end{aligned}$$

where in the second line we have merged the nested optimization over ν and π, see Remark 4.1. To obtain $E_{\text{s},2}(\Lambda)$ all occurrences of λ will again be replaced by independent separate optimizations over Λ. To handle the dependency of the feasible set on λ_s consider the following set:

$$\begin{aligned}\text{SegCoupl}(Y, \mu_1, \mu_2) = \{\pi \in \text{Meas}(X \times Y) :\ & \mu_1 \\ \leq \text{Proj}_{X\sharp}\pi \leq \mu_2 \wedge \text{Proj}_{Y\sharp}\pi \leq \mathcal{L}_Y\Big\}& \qquad (4.12)\end{aligned}$$

Obviously $\text{SegCoupl}\left(Y, (1+\lambda_\text{s})^2 \cdot \mu\right) \subset \text{SegCoupl}\left(Y, (1+\lambda_{\text{s,l}})^2 \cdot \mu, (1+\lambda_{\text{s,u}})^2 \cdot \mu\right)$ as long as $\lambda_{\text{s,l}} \leq \lambda_\text{s} \leq \lambda_{\text{s,u}}$. Then a possible definition of $E_{\text{s},2}$ equivalent to (4.7) is

$$\begin{aligned}E_{\text{s},2\text{a}}(\Lambda) &= \inf_{\pi \in \text{SegCoupl}\left(Y,(1+\lambda_{\text{s,l}})^2\cdot\mu,(1+\lambda_{\text{s,u}})^2\cdot\mu\right)} \\ &\left(\min_{\lambda_\text{s} \in [\lambda_{\text{s,l}}, \lambda_{\text{s,u}}]} \frac{1}{2\,(1+\lambda_\text{s})^2}\right) \int_{X \times Y} \left(\left(\inf_{\lambda \in \Lambda} c_\text{geo}\left(T_\lambda(x), y\right)\right)\right. \\ &\left. + c_\mathcal{F}(f_x, f_y)\right) d\pi(x,y) + \inf_{\lambda \in \Lambda} F(\lambda) + G(\text{Proj}_{Y\sharp}\pi) \qquad (4.13)\end{aligned}$$

where $\lambda_{\text{s,l}}$ and $\lambda_{\text{s,u}}$ are the infimum and supremum of λ_s in Λ. It is easy to see that $E_{\text{s},2\text{a}}$ satisfies Proposition 4.3 w.r.t. $E_{\text{s},1}$. The proof is analogous.

If G is zero the definition of $E_{\text{s},2\text{a}}$ can be improved upon. Consider the following lemma:

Lemma 4.6 *For some cost function c and $m > 0$ let*

$$f(m) = \inf_{\pi \in SegCoupl(Y, m\cdot\mu)} \int_{X \times Y} c(x,y)\, d\pi(x,y). \qquad (4.14)$$

Then $f(m_2)/m_2 \geq f(m_1)/m_1$ for $m_2 > m_1$.

Proof Assume $f(m_2) < (m_2/m_1) \cdot f(m_1)$ for $m_2 > m_1$ and let π_2^* be an optimizer for $f(m_2)$. Then $(m_1/m_2) \cdot \pi_2^*$ is feasible for computation of $f(m_1)$ and one has

$$\frac{m_1}{m_2} \int_{X \times Y} c(x,y)\, d\pi_2^*(x,y) = \frac{m_1}{m_2} f(m_2) < f(m_1) \qquad (4.15)$$

which is a contradiction. □

With the aid of Lemma 4.6 one then finds that the following is a suitable variant of $E_{\text{s},2\text{a}}$:

$$\begin{aligned}E_{\text{s},2\text{b}}(\Lambda) &= \inf_{\pi \in \text{SegCoupl}\left(Y,(1+\lambda_{\text{s,l}})^2\cdot\mu\right)} \frac{1}{2\,(1+\lambda_{\text{s,l}})^2} \\ &\int_{X \times Y} \left(\left(\inf_{\lambda \in \Lambda} c_\text{geo}\left(T_\lambda(x), y\right)\right)\right. \\ &\left. + c_\mathcal{F}(f_x, f_y)\right) d\pi(x,y) + \inf_{\lambda \in \Lambda} F(\lambda) \qquad (4.16)\end{aligned}$$

The advantages over $E_{\text{s},2\text{a}}$ are a tighter scaling factor and a simpler feasible set for the optimal transport term.

Remark 4.7 (Scale mode and alternating optimization). The alternating optimization scheme presented in Sect. 4.1 only works with zero-divergence modes. The hierarchical optimization scheme can be used to extend this to the scale mode. The non-scale coefficients are determined by separate optimization as before, see (4.4b). The new coefficient λ_s^{k+1} and π^{k+1} are jointly determined by global hierarchical optimization, while keeping the other mode coefficients fixed (this replaces (4.4a)). This hierarchical scheme will only go over one degree of freedom and thus be very quick. Again one finds a non-increasing sequence that must eventually converge.

4.3 Graph Cut Relaxation

Both alternating and hierarchical optimization require solving a lot of optimal transport problems. Even with efficient solvers this will quickly become computationally expensive as the size of X and Y or the number of modes increases. If G is non-zero then usually even more so because dedicated optimal transport solvers can no longer be applied directly to compute $E_1(\lambda)$. Therefore, in this section we present a mass-constraint relaxation that, for suitable choice of G, turns computation of $E_1(\lambda)$ into a min-cut problem. This can be solved very fast with dedicated algorithms and therefore the relaxation yields a huge speed-up.

Throughout this section let X and Y be discrete sets, e.g. pixels or super-pixels. The Lebesgue measure on Y is approximated by

$$\mathcal{L}_Y(\sigma) = \sum_{y \in \sigma} m_y \qquad (4.17)$$

for subsets $\sigma \subset Y$, where m_y is the area of super-pixel y. Any $\nu \in \text{SegMeas}(Y, M)$ can then be expressed as

$$\nu(\sigma) = \sum_{y \in \sigma} m_y\, u_\nu(y) \qquad (4.18)$$

for all $\sigma \subset Y$ with some function $u_\nu \colon Y \to [0,1]$. Let G be a total-variation-like local boundary regularizer of ν, expressed in terms of u_ν:

$$G(\nu) = \sum_{(y,y') \in \mathcal{G}} a_{y,y'} \cdot |u_\nu(y) - u_\nu(y')| \tag{4.19}$$

where $\mathcal{G}$ is the set of super-pixel neighbours and $a_{y,y'}$ is a weight that models the likelihood of a boundary between neighbours y and y'. Such weights can be constructed from feature dissimilarity in y, y', from the response of edge detectors and from the length of the boundary.

We now relax the template-marginal constraint from the coupling set $\Pi(\mu, \nu)$ and allow ν to have arbitrary mass. So the feasible set of ν will be

$$\mathrm{SegMeas}(Y) = \left\{ \nu \in \mathrm{Meas}(Y) : \ \nu \leq \mathcal{L}_Y \right\}. \tag{4.20}$$

This is (2.3) without the mass constraint. The 'couplings' π will be taken from the set

$$\hat{\Pi}(\nu) = \left\{ \pi \in \mathrm{Meas}(X \times Y) : \ \mathrm{Proj}_{Y\sharp}\pi = \nu \right\}. \tag{4.21}$$

Merging optimizations (see Remark 4.1) yields the feasible set

$$\mathrm{SegCoupl}(Y) = \left\{ \pi \in \mathrm{Meas}(X \times Y) : \ \mathrm{Proj}_{Y\sharp}\pi \leq \mathcal{L}_Y \right\}. \tag{4.22}$$

The relaxed equivalent of E_1 (4.6) that we consider in this section is

$$E_{\mathrm{r},1}(\lambda) = \inf_{\pi \in \mathrm{SegCoupl}(Y)} \frac{1}{2} \int_{X \times Y} \Big(c_{\mathrm{geo}}\big(T_\lambda(x), y\big) + c_{\mathcal{F}}(f_x, f_y) \Big) \, d\pi(x,y) + F(\lambda) + G(\mathrm{Proj}_{Y\sharp}\pi). \tag{4.23}$$

Let π^* be an optimizer of $E_{\mathrm{r},1}(\lambda)$ for some configuration λ. If $(\mathrm{Proj}_{Y\sharp}\pi^*)(y) > 0$ for some $y \in Y$, this mass will come from the cheapest $x \in X$ for this y, since there is no longer any constraint on the mass on X. The linear matching in the first term simplifies to a nearest neighbour matching for each $y \in Y$. This implies that the minimization in (4.23) over $\pi \in \mathrm{SegCoupl}(Y)$ can be simplified to a minimization over $\nu \in \mathrm{SegMeas}(Y)$. Therefore (4.23) is equivalent to

$$E_{\mathrm{r},1}(\lambda) = \inf_{\nu \in \mathrm{SegMeas}(Y)} \frac{1}{2} \sum_{y \in Y} c_{\min}(y, \lambda)\, \nu(y) + F(\lambda) + G(\nu) \tag{4.24}$$

with

$$c_{\min}(y, \lambda) = \min_{x \in X} \left(c_{\mathrm{geo}}\big(T_\lambda(x), y\big) + c_{\mathcal{F}}(f_x, f_y) \right). \tag{4.25}$$

We express now ν in terms of u_ν, see (4.18), and plug in the form of the regularizer G (4.19). This yields

$$E_{\mathrm{r},1}(\lambda) = \inf_{u: Y \to [0,1]} \frac{1}{2} \sum_{y \in Y} c_{\min}(y, \lambda) \cdot m_y \cdot u(y) + F(\lambda) + \sum_{(y,y') \in \mathcal{G}} a_{y,y'} \cdot |u(y) - u(y')|. \tag{4.26}$$

For fixed λ this is a convex formulation of the max-flow / min-cut problem with nodes Y and edges $\mathcal{G}$. The edge-weight between $y \in Y$ and the sink is given by $c_{\min}(y, \lambda) \cdot m_y$ and the weights of the edges between $y, y' \in Y$ by $a_{y,y'}$. This problem can be solved very efficiently by dedicated algorithms, see for example [6].

Remark 4.8 (Optimization of $E_{\mathrm{r},1}$). Both the alternating method and the hierarchical scheme, Sects. 4.1 and 4.2, can be applied directly to the optimization of $E_{\mathrm{r},1}$. The sequence equivalent to (4.4) will provide a non-increasing converging sequence of energies. Since the dependence of the feasible set on the mass of μ has disappeared, it can also be extended to the scale mode. Also, handling the scale mode in the hierarchical scheme is simplified.

Functional (4.26) can be interpreted as a binary Markov random field (MRF) with labels fore- and background ($u \in \{1, 0\}$) and a latent object configuration variable λ. Such enhanced MRFs have been used in [21] with the latent variables describing layered pictorial structures and in [37] with graph-based shape models. Optimization of a general class of such models via branch and bound has been discussed in [23]. A main difference of the approach presented here and [23] is that the shape variations are not captured implicitly in the hierarchical cluster of sample shapes but explicitly and smoothly in the set of learned Wasserstein modes.

5 Numerical Examples

We will now present some numerical examples for joint image segmentation and shape matching with Wasserstein modes. The scope of these examples is to transparently show the key properties of the functional (geometric invariance, response to noisy data etc.) and to demonstrate its applicability to different types of geometric data and features.

5.1 Setup and Implementation Details

Setting up the Model. As discussed in Sect. 3.3 the functional component F, modelling the distribution of the deformation parameter λ (c.f. (3.5)), was not depending on the

λ-entries that describe translation, rotation and scale. For the statistical modes we modelled a simple Gaussian as given by (3.16). The weight γ was set to a small value, i.e. we 'trusted' the data for small deformations and mainly wanted to keep the deformations from becoming too large, where the linear deformation model does no longer work very well.

The number of used modes ranged between 3 and 8 for branch and bound, up to about 14 for the alternating scheme. As discussed in Sect. 4.2, for the initial covering L of the parameter space for λ, we used a grid of n-dimensional hypercubes: for the translation components ranging over the area of the image, for rotation and scale within the limits where the numerical approximation is valid and for the statistical modes depending on the observed standard deviations during learning.

A very important parameter in the functional is the relative weight between the geometric and the appearance cost function, c_{geo} and $c_{\mathcal{F}}$. When the appearance features are very noisy, we tend to put more trust on the geometric component and thus the predefined deformation modes. For very reliable data we may accept a previously unknown deformation to better match the observed features. Some intuition on how to choose this relative weight may be gained from Fig. 8.

Optimization Algorithms. In most experiments numerical optimization was carried out in two steps, starting with branch and bound over the modes with largest deformations, followed by alternating optimization over all modes (see Remark 4.5). As pointed out in Remark 3.2 continuous solvers cannot be applied since the marginal ν is unknown. Therefore we rely on discrete algorithms. For numerical optimization of $E_2(\Lambda)$ we implemented two different methods:

- For $G = 0$, i.e. in the absence of an additional segmentation term on the marginal ν (c.f. (3.5)), the functional $E_2(\Lambda)$ (4.7) can be evaluated by using a dedicated optimal transport solver. For this we wrote a c + + implementation of the Hungarian method [20].
- When G is a discrete total-variation-like local regularity prior (see Sect. 4.3, Eq. (4.19)) evaluation of $E_2(\Lambda)$ can be written as a linear program, which we solved with `CPLEX`.

The top level, i.e. everything except for the calls to optimize $E_2(\Lambda)$ was implemented in Mathematica.

In practice we used the first variant for the branch and bound stage and the second variant for the subsequent alternating optimization stage. The reasoning behind this is that the total variation of a segmentation depends mostly on its local properties and can be vary significantly without altering its global configuration, which is what we look for during the branch and bound optimization. TV is then added during the 'fine-tuning' in the alternating stage.

Reducing Complexity in Practice. To reduce computational complexity, we sampled the cost function $c_{\text{geo}}(x, y) + c_{\mathcal{F}}(f_x, f_y)$ for fixed x only at positions y close to x. When y is very far from x the high geometric cost will make the assignment very unlikely. The cut-off radius around x is chosen according to the range of $c_{\mathcal{F}}$ and the size of the mode parameter set Λ during branch and bound. Global optimality of the sub-sampled cost-function w.r.t. the dense model can be checked by introducing 'overflow' variables with suitable assignment costs for each x: as long as no mass is put onto these overflow variables, the optimizer of the reduced model is also globally optimal in the dense model.

Computational Complexity and Runtime. Although we only used experimental code, which was far from being optimized for performance we briefly comment on the observed running-times to give the reader a general idea of the applicability. Experiments were performed on a standard desktop computer with an Intel Core i7 processor at 3.4 GHz and 16 GB RAM. The branch and bound scheme, which is the computationally most demanding part, was parallelized over the processor cores. The alternating optimization is much less demanding and consequently converges much faster. The discrete templates had several 100 points, the discrete images, super-pixel segmentations, etc. several 1,000 points.

For the branch and bound scheme the running time is determined by how many of the tree of bounds at different scales have to be explored until a minimizer is found. This number is sensitive to several factors: it grows exponentially with the number of degrees of freedom. Also, it depends on the specific problem instance and how well the global optimum is pronounced. In the presence of strong noise or multiple similarly good minima the scheme will naturally take longer as in a problem with only one distinct solution. Consequently it is not really possible to accurately estimate the number of required bounds beforehand, i.e. to give an overall expected complexity estimate of the branch and bound scheme.

During our experiments we observed running times from under a minute for 3–4 modes on 'easy problems' up to about a day for 7–8 modes on very noisy and large instances. Instances shown in this section were mostly set up such that branch and bound would take 10 min at most.

Of course the running time also depends strongly on the problem dimensions. Fortunately, the flexible mathematical framework provides means for reducing the problem dimensions easily by working for example on an over-segmentation with super-pixels instead of on the full pixel grid. The loss of resolution can often be compensated for by adding a local regularizer.

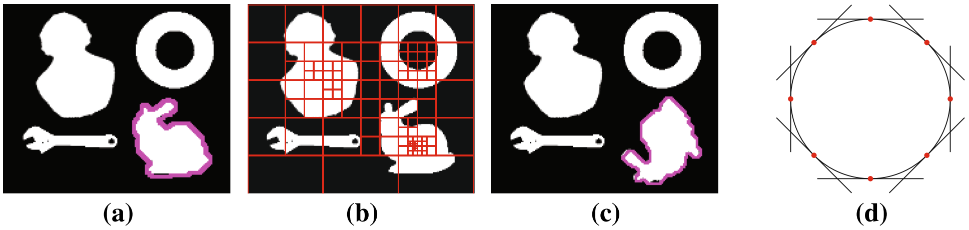


Fig. 4 Shape location with branch and bound. We are searching for a bunny among a collection of other shapes via branch and bound. **a** Illustration of $c_{\mathcal{F}}(f_x, f_y)$, white indicating foreground affinity. The optimal segmentation is given by the purple line. **b** The covering set L (4.10) upon convergence of branch and bound, projected onto the two translation components, shown relative to the query image. Very dissimilar shapes such as the wrench can be ruled out at a coarse level while more similar shapes such as the bust can only be discarded on finer scales. The grid is finest at the true location of the bunny. **c** Modified problem with the bunny rotated by a large angle. Such large angles cannot be covered by the rotation mode, see (3.11) and its discussion. Instead, one can use multiple support points on the shape manifold (sketched in **d**), each equipped with a local rotation mode, and integrate them into the branch and bound scheme (Color figure online)

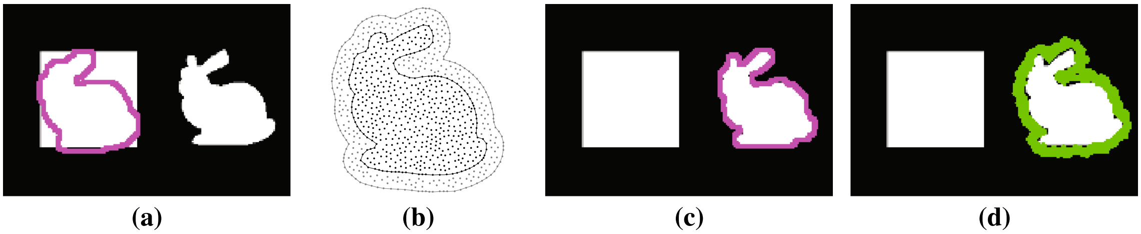


Fig. 5 Background modelling. **a** Naïve segmentation without modelling the image background: sometimes it is then the optimal configuration to 'immerse' the sought-after shape into a large blob of false-positive detections. **b** The shape template: to solve this problem, we can model a small area of background (*gray*) around the boundary of the object (*black*). **c** Optimal segmentation when the background around the object is modelled. **d** Region which is assigned to the explicitly modelled background (Color figure online)

5.2 Numerical Results

We start with some synthetic experiments to transparently illustrate different properties of the functional. For these experiments the feature cost function $c_{\mathcal{F}}(f_x, f_y)$ was chosen to be constant w.r.t. x, i.e. every template point expects the same features and the template has a homogeneous appearance. This corresponds to a classifier that tries to locally asses for each pixel whether it is part of the fore- or background.

Branch and Bound. A shape model of a bunny is learned from several different views. The subsequent task is then to find a novel view (within the range of the training views) among a collection of different shapes. Branch and bound was used to optimize over translations, rotation and scale of the object. On these degrees of freedom the alternating scheme is prone to getting stuck in a poor local minimum, if initialized on the wrong shape. Afterwards the alternating scheme was applied to account for non-isometric variations due to perspective. Additionally it is shown how the rotation invariance can be extended to large angles. The results of this experiment are illustrated in Fig. 4.

Background Modelling. Note that in order to locate the bunny correctly, we sometimes also need to model the image background in some way. This can be done implicitly by ensuring that the boundary of the foreground is aligned with detected contours in the image via a weighted TV-like term through $G(\nu)$ in (3.5). A more explicit approach is to extend the template to include a small region 'looking like background' around the foreground (see Sect. 3.5). This is demonstrated in Fig. 5.

More examples on detecting objects in a noisy environment and on restoring shapes from distorted detections are given in Figs. 6 and 7.

Interaction of Regularizers. Now let us study the interaction between the different components of the functional. Let G be the discrete total variation of ν (4.19). For now we ignore deformations and simply take a fixed template. That is we consider the following functional:

$$E(\nu) = \inf_{\pi \in \Pi(\mu,\nu)} \int_{X \times Y} \Big(\|x - y\|^2 + \tau \cdot c_{\mathcal{F}}(f_x, f_y) \Big) \, d\pi(x, y) + \sigma \cdot G(\nu) \tag{5.1}$$

where we have introduced weights τ and σ. In Fig. 8 it is illustrated how the optimal segmentations depend on τ and σ in the presence of different types of noise.

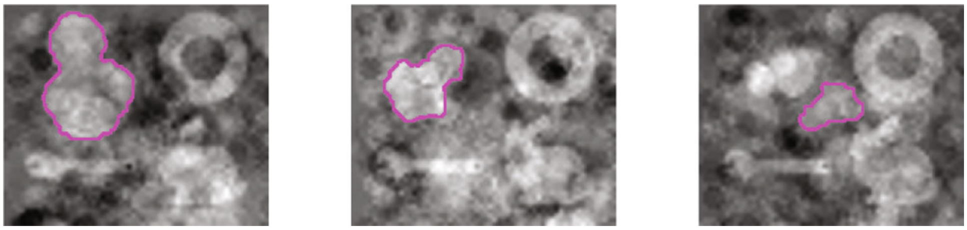


Fig. 6 Locating shapes in a noisy environment. We are looking for the bust of Beethoven in a picture with non-local noise and other shapes present. In the first two examples the shape is correctly identified. In the third example, the true bust is missed, because it is rather small and instead a chunk of false-positive noise is segmented (Color figure online)

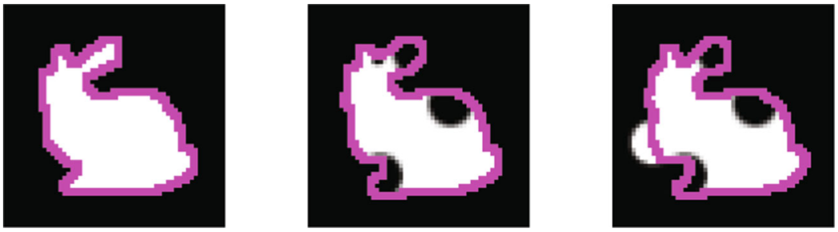


Fig. 7 Restoring distorted shapes. By aid of the template geometry non-local noise, e.g. partial occlusion and false-positive detections can be recognized and the segmentation retains the true sought-after shape (Color figure online)

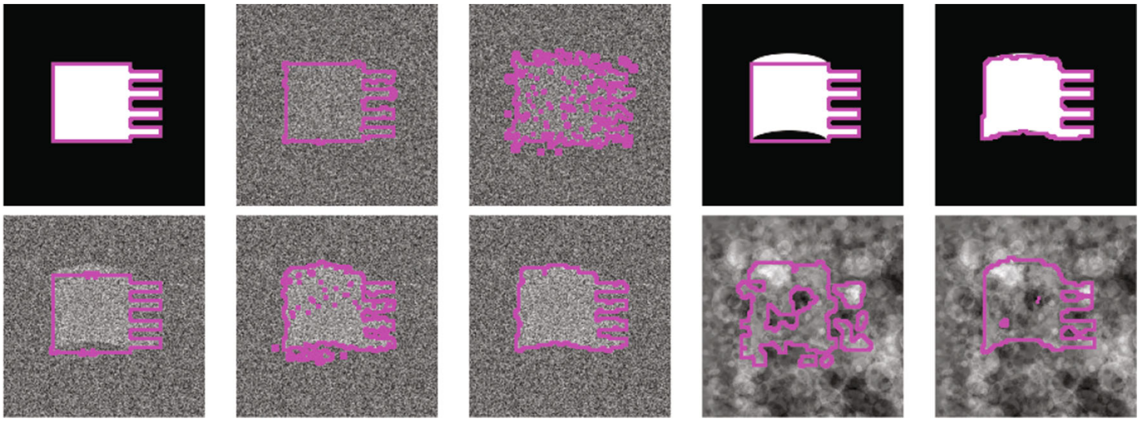


Fig. 8 Interaction of Regularizers. *Top row, from left to right:* (1) Clean problem with object formed exactly like template. (2) Local Gaussian noise, low feature-cost weight τ, $\sigma = 0$, i.e. the optimal matching is dominated by the geometric cost component. (3) Same problem as before, but with a high τ: now the local noise severely affects the segmentation, which becomes very irregular. (4) An unknown deformation is encountered (not described by a known deformation mode). With low τ it is ignored. (5) With a higher τ the optimal segmentation locally adapts to the unknown deformation. *Bottom row:* (1) Unknown deformation with local noise and low τ: now the trick to simply increase τ (2) to adapt for the unknown deformation does no longer work, as the local noise is distorting the segmentation. (3) The problem can be solved by adding a local boundary regularizer ($\sigma > 0$): it helps to distinguish between the local Gaussian noise and the non-local unknown deformation. So the optimal segmentation ignores the former but adapts to the latter. (4) The same trick does not work with non-local noise: now unknown deformation and non-local noise cannot be separated and the optimal segmentation becomes faulty. (5) Adding the deformation as a Wasserstein mode helps to approximately find the object even in this noisy scenario, also thanks to the robustness of the globally optimal branch and bound scheme (Color figure online)

Alternating Optimization. In Fig. 9 the behaviour of the alternating optimization scheme is elucidated. In particular it becomes apparent how in noisy problems the scheme easily gets stuck in poor local minima. This is a general problem of local optimization methods and proves the importance of the globally optimal branch and bound scheme to provide a proper initial starting point.

Super-pixels. An important feature of functional (3.5) is that its discrete version readily encompasses a wide range of data structures. As the computational complexity strongly depends on the size of the discretizations of X and Y it may be reasonable to apply the functional not directly to the pixel level but to a coarser over-segmentation as for example provided by super-pixels. Some examples with the class

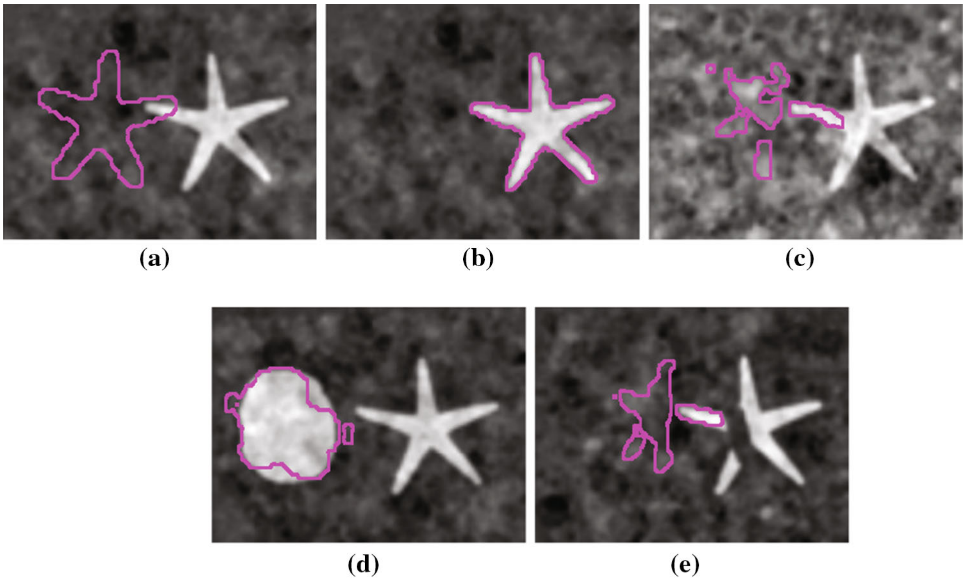


Fig. 9 Alternating Optimization. The fundamental limitations of the local optimization scheme become apparent in this experiment. **a** Initial position, some overlap with true segmentation is given. **b** Upon convergence the true shape has been located. **c** Same scenario but with a higher noise: now the alternating scheme gets stuck along the way. **d** A large, but not starfish-shaped blob on the left by mistake attracts the template. **e** The partial occlusion of the shape obstructs the convergence. The local scheme has no way of knowing 'that the starfish continues' beyond the occlusion (Color figure online)

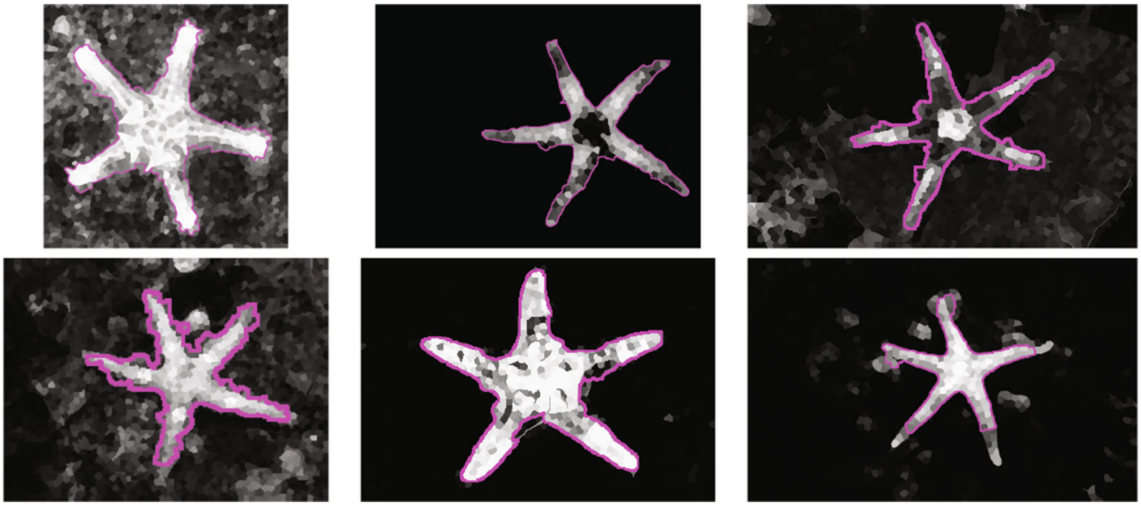


Fig. 10 Application to super-pixel images. The numerical framework extends seamlessly to super-pixel images. A simple local classifier based on color was applied to super-pixel images of starfish. The classifier was intentionally designed to yield partially faulty results. With simultaneous matching and segmentation, locally faulty detections can be corrected for: false-positive clutter is ignored, missing parts are restored. On the *bottom-right* an example is given where the deformation modes are not flexible enough to adapt to the true object shape (Color figure online)

'starfish' are given in Fig. 10. Figure 11 shows some of the involved non-isometric deformations to illustrate the range of the linear modes model and also one example where the limit of the linear expansion has been reached.

In Fig. 12 the scale invariance of the approach is demonstrated by actually deliberately breaking it. The same functional is optimized twice, but with a different prior on the allowed object scale. Depending on the admissible scale, once the large and once the small clownfish is segmented. Such a task can only be solved with global optimization techniques.

Inhomogeneous $c_{\mathcal{F}}$. So far we have only considered the case where $c_{\mathcal{F}}(f_x, f_y)$ was constant w.r.t. x. However, computationally there is no increase in complexity if we pick a more general feature cost. The potential of this additional freedom is now demonstrated on an example with the UIUC database (see for example [1]). This is a set of gray level side views of parking cars. Locating these cars cannot be approached with a homogeneous foreground / background detector, as no consistent separation based on local appearance features seems to be possible.

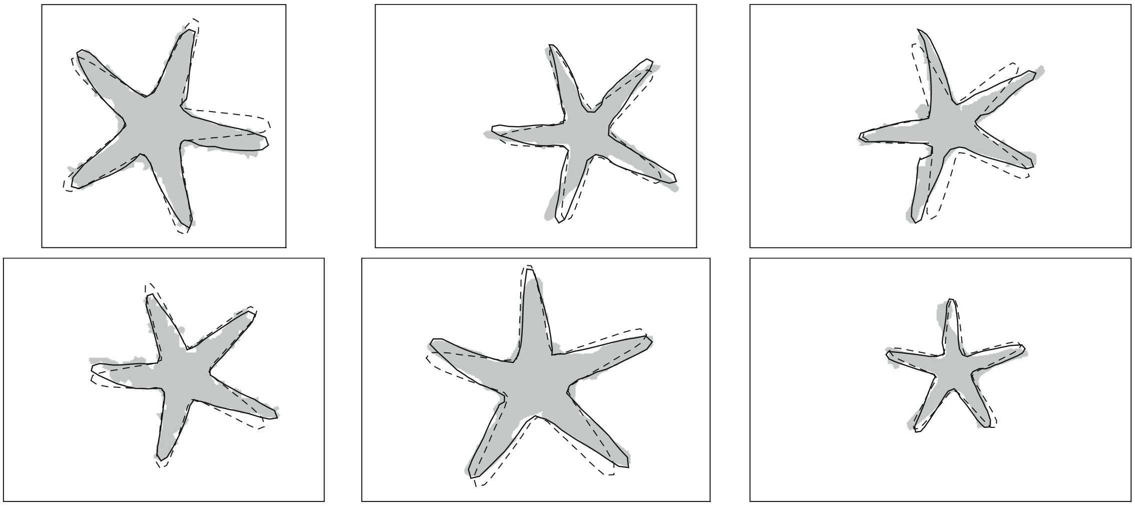


Fig. 11 Range of linear deformation model. For the segmentations in Fig. 10 we illustrate here in the same order the relative configuration of the template after translation and rotation (*dashed lines*), the fully transformed template (*black lines*) and the segmentation (*gray shading*). The experiments used three isometric (translation + rotation) and ten statistical modes. One can see that substantial changes in shape can be encoded by the linear modes. On the top-right an example is shown where the deformation coefficients λ have become too large and the shape looks distorted

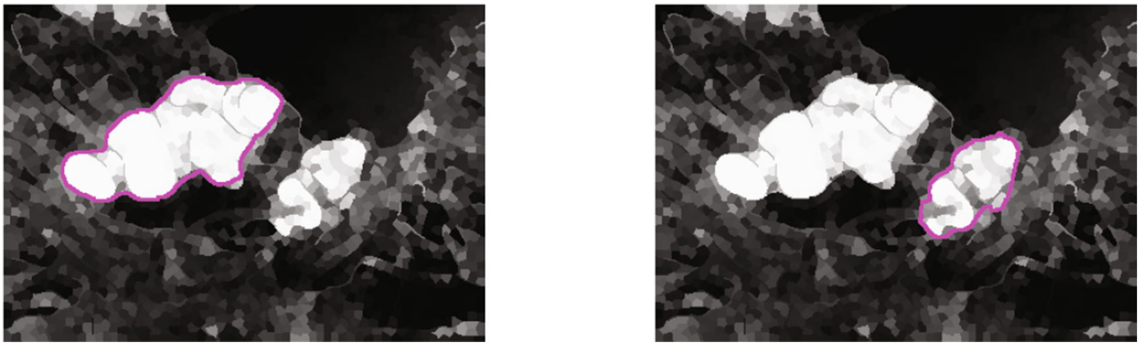


Fig. 12 Scale invariant segmentation. Similar to the starfish experiment, a super-pixel image of two clownfish is to be segmented, based on an imperfect local color classifier. With the aid of a shape prior, by setting a preferred range of object scales, but leaving the rest of the approach scale invariant, depending on the choice, both the large and the small fish are correctly located. Note that this example also requires proper modelling of the object boundary (see Fig. 5) (Color figure online)

Therefore we now learn local detectors for each point of the template X separately and based on these compute an inhomogeneous $c_{\mathcal{F}}$. As features we use local histograms of the image color and its gradient. We compute assignments between the learned template and the training cars (both shapes fixed, only geometric, no appearance cost). Based on these assignments we extract for each template point x the collection of expected features f_x. Then, on a test image Y we compare for each super-pixel $y \in Y$ its histogram of features f_y with the distribution of expected features f_x on each template point via an optimal transport based histogram distance (see e.g. [28]). These comparison costs were used as costs $c_{\mathcal{F}}(f_x, f_y)$.

We want to emphasize at this point that we do in no way champion this particular choice of features and this choice does not constitute a part of our presented framework. We merely seek to provide a transparent set-up to demonstrate the benefit of locally adaptive template appearance without obstruction through more complicated feature acquisition and processing.

Figure 13 gives an impression of the functions $c_{\mathcal{F}}(f_x, f_y)$ obtained in this way. Obviously, for a single template point $x \in X$ the associated cost is very noisy and not very informative. We can thus only hope that through the combination of all template pixels and the knowledge about their relative spatial arrangement we can identify the positions of the cars.

Since the variation of the shapes of the cars is small we only consider translations during branch and bound for locating the cars. Geometric flexibility beyond that is provided by the optimal transport matching. In this way on 10 out of 15

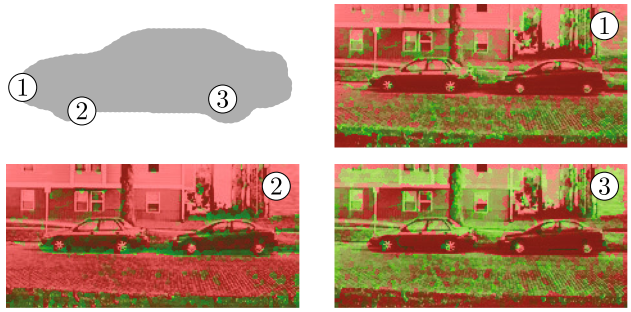


Fig. 13 Inhomogeneous appearance model. For each template pixel a local appearance model was learned. *Top left:* template X with three selected (super-)pixels $\{x_i\}_i$. *Top right, bottom row:* costs $c_{\mathcal{F}}(x_i,\cdot)$ for the three selected pixels. The appearance cost of single pixels is not very informative. Only by combining costs from all template pixels and their relative spatial position enables one to find the objects (Fig. 14) (Color figure online)

test images the global optimum correctly corresponded to a car (some images show multiple cars). As baseline we performed a simple Hough transform which failed to correctly locate any car. Figure 14 gives some example cases and also illustrates a failed case.

A similar experiment was performed in [23]. There the main focus was on modelling the boundary of the cars whereas here we concentrate on its region. Both approaches can incorporate both cues from the object interior as well as its boundary. In Sect. 4.3 it was discussed how [23] is closely related to the graph-cut relaxation of our functional. The most significant difference is how in our approach the geometric variability is explicitly modelled by a linear space of modes whereas in [23] it is implicitly encoded in a hierarchical clustering.

Adaptive $c_{\mathcal{F}}$. We have already mentioned in Remark 3.4 that the Wasserstein modes can also be extended beyond geometric variations to the feature component. This is of particular use when an expected feature is known to change under a certain geometric transformation. For example the orientation of an expected gradient changes with rotation. More generally, a vector valued feature f_x will have to be transformed by

$$DT_\lambda(x) = \mathrm{id} + \sum_{i=1}^{n} \lambda_i \, Dt_i(x), \tag{5.2}$$

the Jacobian of the applied transformation, to preserve it's 'relative orientation' within the template, and we see that this yields a linear deformation on the feature space.

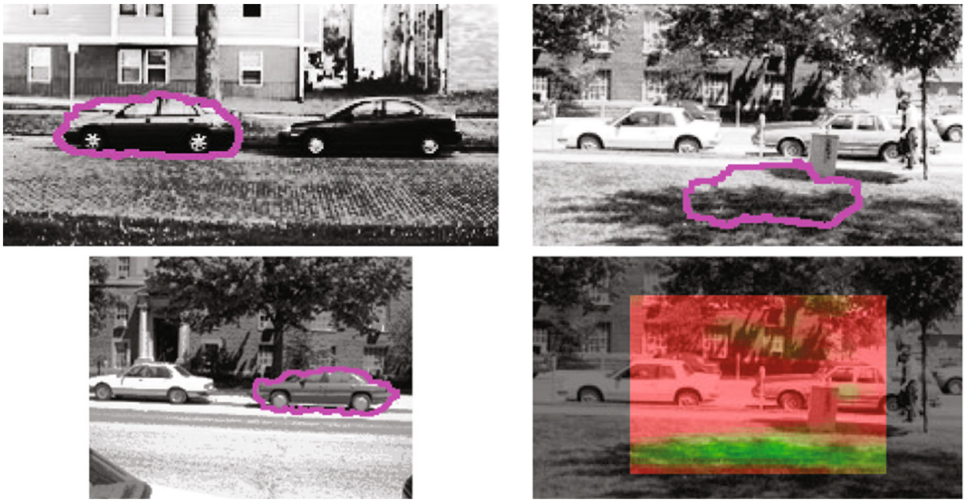
Here we provide a simple example to point out the potential of this flexibility. We now assume that both location and expected feature of a template point vary with the transformations. We model this by linearly expanding $c_{\mathcal{F}}$ in λ around the origin. That is we choose (c.f. (3.7–3.9)):

$$\begin{aligned}\hat{c}\big(\hat{T}_\lambda(x),(y,f_y)\big) &= c_{\mathrm{geo}}\big(T_\lambda(x),y\big) + c_{\mathcal{F}}(f_x,f_y)\\ &\quad + \sum_{i=1}^{n} \lambda_i \cdot c_{\mathcal{F},i}(f_x,f_y)\end{aligned} \tag{5.3}$$

where $c_{\mathcal{F},i}(f_x,f_y)$ is the partial derivative of the feature component of $\hat{c}\big(\hat{T}_\lambda(x),(y,f_y)\big)$ w.r.t. λ_i evaluated at zero (thus giving the first order change along the feature component of $\hat{t}_i$). Both discussed optimization schemes can easily be adapted to this extension.

As a toy example we will be looking for apples. Unripe, small apples are assumed to be green, ripe, large apples should have a reddish color. That is, the expected color varies with size (Naturally the apparent size of an apple on the image depends strongly on the distance from the camera. But we will generously overlook this for the sake of the demonstration.) The results of our search for fruit are illustrated in Fig. 15.

Fig. 14 Locating cars with a spatially inhomogeneous appearance model. *Left column.* Two successful examples of locating a car within the test image. *Right column. Top*: A failed example. *Bottom*: plot of the matching cost depending on translation. Apparently the chosen features are too simple: the shady patch of lawn in the foreground has by far the best cost. Note however that on the right car there is a distinct local minimum (Color figure online)



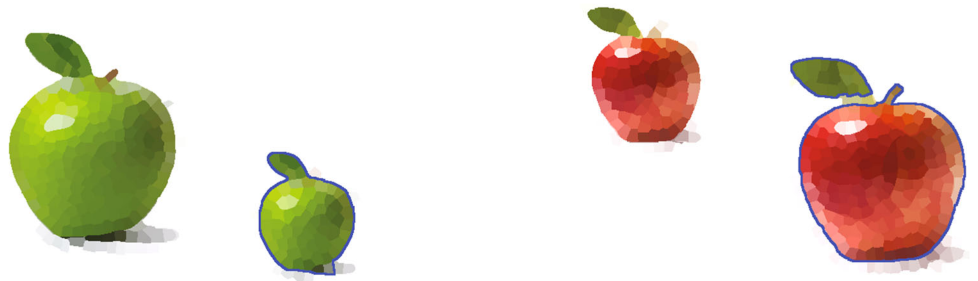


Fig. 15 Locating apples with dynamic appearance. We are looking for apples in the test image. According to our model, small apples should appear *green* (unripe) and large apples *reddish*. This change in appearance, depending on the geometric state, can be encoded by a Wasserstein mode that extends to the feature cost function. Consequently, the small *green* and the large *red* apple are detected, while the 'implausible' large *green* and small *red* apple are discarded (Color figure online)

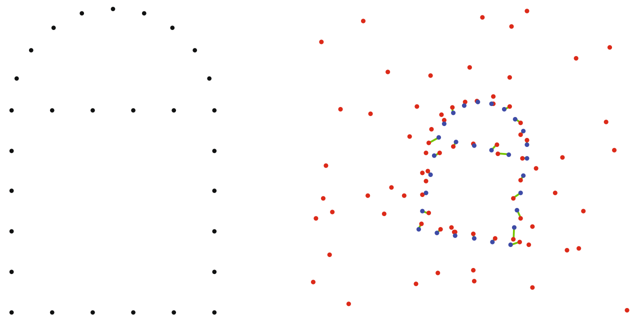


Fig. 16 Segmenting and matching on point clouds. *Left:* the template, a schematic 'gate'. *Right:* the original shape is subjected to perspective transformations (foreshortening, rotation, scale), noise and additional, noisy observations are added. With branch and bound the original shape is detected. This could be applied to matching sparse interest points on images (Color figure online)

Point Clouds. Last but not least we want to further illustrate the flexibility of the numerical framework by applying it to a scenario with point clouds. This is relevant when one does not deal with dense images but only with sparse interest points. We give a transparent, synthetic example in Fig. 16.

6 Conclusion

We have presented a functional for simultaneous image segmentation and shape matching to correctly locate and segment objects within images under noisy conditions.

Matching is based on optimal transport with a cost function that combines geometric plausibility with consistency of appearance features. Through the convex Kantorovich formulation in terms of coupling measures the functional can naturally be combined with other segmentation terms known from convex variational image segmentation. To implement geometric invariances and to account for non-isometric shape variations we introduced additional degrees of freedom, drawing from the Riemannian structure of the 2-Wasserstein space. Through an equivalence relation of the class of shape measures with closed contours this enabled us to introduce well established shape analysis tools from the contour regime into the segmentation approach while remaining in the measure representation.

While the resulting functional is non-convex, this non-convexity is constrained to a low dimensional variable which allowed us to devise an adaptive convex relaxation on which a globally optimal branch & bound optimization scheme could be constructed. Alternatively, a faster but only locally optimal alternating optimization scheme was discussed. While it seems impractical to run the branch and bound scheme on a high number of deformation modes, it still provides a consistent way to find good initializations for the alternating scheme, thus overcoming a severe problem in many other segmentation/matching approaches. Determining a good initial guess and the subsequent 'fine tuning' are based on the very same model and only differ in the application of the optimization scheme. To reduce numerical complexity, a graph-cut relaxation was discussed.

In Sect. 5 we presented a series of numerical examples to demonstrate various aspects of the approach. The basic behaviour of the branch and bound scheme was illustrated as well as the limitations of the alternating scheme. It was shown how the location and shape of the optimal segmentations depends on noise and how different kinds of noise can at least partially be handled by properly choosing the weights between the different terms of the functional. We put a particular focus on illustrating the flexibility in both spatial data structure (pixels, super-pixels, point clouds) as well as in incorporating different types of knowledge on the object appearance (spatially varying, adaptive to deformations).

In the presented state a major limitation of the functional is the linearity of the modes: this makes it difficult to handle large deformations. In this respect other approaches such as the LDDMM framework [4,15,39] are already much further developed, yet focus on smooth registration mappings without addressing variational segmentation simultaneously and explicitly. On the other hand we notice that in terms of handling local feature data this approach is similarly flexible (compare for example with [9]). Also, we consider the branch and bound scheme as an important step towards coherently solving the initialization problem.

Future work should therefore focus on making the deformations more flexible and powerful while trying to retain the ability to obtain robust initializations.

Acknowledgments This work was supported by the DFG, Grant GRK 1653.

References

1. Agarwal, S., Awan, A., Roth, D.: Learning to detect objects in images via a sparse, part-based representation. IEEE Trans. Pattern Anal. Mach. Intell. **26**(11), 1475–1490 (2004)
2. Agueh, M., Carlier, G.: Barycenters in the wasserstein space. SIAM J. Math. Anal. **43**(2), 904–924 (2011)
3. Ambrosio, L., Gigli, N.: A user's guide to optimal transport. In: Modelling and Optimisation of Flows on Networks. Lect. Not. Math., vol. 2062, pp. 1–155. Springer, Heidelberg (2013)
4. Beg, M.F., Miller, M.I., Trouvé, A., Younes, L.: Computing large deformation metric mappings via geodesic flows of diffeomorphisms. Int. J. Comput. Vis. **61**(2), 139–157 (2005)
5. Berkels, B., Fletcher, T., Heeren, B., Rumpf, M., Wirth, B.: Discrete geodesic regression in shape space. In: Energy Minimization Methods in Computer Vision and Pattern Recognition (EMMCVPR 2013), pp. 108–122 (2013)
6. Boykov, Y., Kolmogorov, V.: An experimental comparison of min-cut/max-flow algorithms for energy minimization in computer vision. IEEE Trans. Pattern. Anal. Mach. Intell. **26**(9), 1124–1137 (2004)
7. Bronstein, A.M., Bronstein, M.M., Kimmel, R.: Efficient computation of isometry-invariant distances between surfaces. SIAM J. Sci. Comput. **28**, 1812–1836 (2006)
8. Chan, T.F., Esedoglu, S., Nikolova, M.: Algorithms for finding global minimizers of image segmentation and denoising models. SIAM J. Appl. Math. **66**(5), 1632–1648 (2006)
9. Charon, N., Trouvé, A.: Functional currents: a new mathematical tool to model and analyse functional shapes. J. Math. Imaging Vis. **48**(3), 413–431 (2014)
10. Charpiat, G., Faugeras, O., Keriven, R.: Approximations of shape metrics and application to shape warping and empirical shape statistics. Found. Comput. Math. **5**(1), 1–58 (2005)
11. Cohen, S.D., Guibas, L.J.: The earth mover's distance under transformation sets. In: International Conference on Computer Vision (ICCV 1999), pp. 1076–1083 (1999)
12. Cremers, D., Kohlberger, T., Schnörr, C.: Shape statistics in kernel space for variational image segmentation. Pattern Recognit. **36**(9), 1929–1943 (2003)
13. Cuturi, M., Doucet, A.: Fast computation of wasserstein barycenters. In: International Conference on Machine Learning (2014)
14. Gilboa, G., Osher, S.: Nonlocal operators with applications to image processing. Multiscale Model. Simul. **7**(3), 1005–1028 (2008)
15. Glaunes, J., Trouve, A., Younes, L.: Diffeomorphic matching of distributions: a new approach for unlabelled point-sets and submanifolds matching. In Computer Vision and Pattern Recognition (CVPR 2004), vol. 2, pp. 712–718 (2004)
16. Haker, S., Zhu, L., Tannenbaum, A., Angenent, S.: Optimal mass transport for registration and warping. Int. J. Comput. Vis. **60**, 225–240 (2004)
17. Heeren, B., Rumpf, M., Wardetzky, M., Wirth, B.: Time-discrete geodesics in the space of shells. Comput. Gr. Forum **31**(5), 1755–1764 (2012)
18. Klodt, M., Cremers, D.: A convex framework for image segmentation with moment constraints. In: International Conference on Computer Vision (ICCV 2011), pp. 2236–2243 (2011)
19. Kriegl, A., Michor, P.W.: The Convenient Setting of Global Analysis. Mathematical Surveys and Monographs, vol. 53. American Mathematical Society, Providence (1997)
20. Kuhn, H.W.: The Hungarian method for the assignment problem. Naval Res. Logist. **2**, 83–97 (1955)
21. Kumar, M.P., Torr, P.H.S., Zisserman, A.: OBJ CUT. In Computer Vision and Pattern Recognition (CVPR 2005), vol. 1, pp. 18–25 (2005)
22. Lellmann, J., Schnörr, C.: Continuous multiclass labeling approaches and algorithms. SIAM J. Imaging Sci. **4**(4), 1049–1096 (2011)
23. Lempitsky, V., Blake, A., Rother, C.: Image segmentation by branch-and-mincut. In: European Conference on Computer Vision (ECCV 2008), pp. 15–29 (2008)
24. Lott, J.: Some geometric calculations on Wasserstein space. Commun. Math. Phys. **277**, 423–437 (2008)
25. Mémoli, F.: Gromov-Wasserstein distances and the metric approach to object matching. Found. Comput. Math. **11**, 417–487 (2011)
26. Michor, P.W., Mumford, D.: Riemannian geometries on spaces of plane curves. J. Eur. Math. Soc. **8**(1), 1–48 (2006)
27. Mio, W., Srivastava, A., Joshi, S.: On shape of plane elastic curves. Int. J. Comput. Vis. **73**(3), 307–324 (2007)
28. Pele, O., Werman, W.: Fast and robust Earth Mover's Distances. In: International Conference on Computer Vision (ICCV 2009) (2009)
29. Pock, T., Chambolle, A., Cremers, D., Bischof, H.: A convex relaxation approach for computing minimal partitions. In: Computer Vision and Pattern Recognition (CVPR 2009), pp. 810–817 (2009)
30. Schmitzer, B., Schnörr, C.: Weakly convex coupling continuous cuts and shape priors. In: Scale Space and Variational Methods (SSVM 2011), pp. 423–434 (2012)
31. Schmitzer, B., Schnörr, C.: Contour manifolds and optimal transport. http://arxiv.org/abs/1309.2240 (2013). preprint
32. Schmitzer, B., Schnörr, C.: Modelling convex shape priors and matching based on the Gromov-Wasserstein distance. J. Math. Imaging Vis. **46**(1), 143–159 (2013)
33. Schmitzer, B., Schnörr, C.: Object segmentation by shape matching with Wasserstein modes. In: Energy Minimization Methods in Computer Vision and Pattern Recognition (EMMCVPR 2013), pp. 123–136 (2013)
34. Sundaramoorthi, G., Mennucci, A., Soatto, S., Yezzi, A.: A new geometric metric in the space of curves, and applications to tracking deforming objects by prediction and filtering. SIAM J. Imaging Sci. **4**(1), 109–145 (2011)
35. Villani, C.: Optimal Transport: Old and New, vol. 338. Grundlehren der mathematischen Wissenschaften. Springer, Berlin (2009)
36. Wang, W., Slepčev, D., Basu, S., Ozolek, J.A., Rohde, G.K.: A linear optimal transportation framework for quantifying and visualizing variations in sets of images. Int. J. Comput. Vis. **101**, 254–269 (2012)
37. Yangel, B., Vetrov, D.: Learning a model for shape-constrained image segmentation from weakly labeled data. In: Energy Minimization Methods in Computer Vision and Pattern Recognition (EMMCVPR 2013), pp. 137–150 (2013)
38. Younes, L., Michor, P.W., Shah, J., Mumford, D.: A metric on shape space with explicit geodesics. Rend. Lincei Mat. Appl. **9**, 25–57 (2008)
39. Younes, L.: Shapes and Diffeomorphisms. Applied Mathematical Sciences, vol. 171. Springer (2010)

Bernhard Schmitzer obtained an M.Sc. in theoretical physics at Imperial College London (2010) and a doctorate in applied mathematics from Heidelberg University (2014). In Heidelberg he was a member of the Image and Pattern Analysis Group and the Research Training Group Probabilistic Graphical Models and Applications in Image Analysis, funded by the German Research Foundation (DFG). As of October 2014 he is a post-doctoral researcher at the CEREMADE at Paris-Dauphine University.

Christoph Schnörr received his degrees from the Technical University of Karlsruhe and the University of Hamburg. He is currently heading the Image and Pattern Analysis Group at the Faculty of Mathematics and Computer Science of the Heidelberg University, coordinating a Research Training Group on Probabilistic Graphical Models and Applications in Image Analysis, funded by the German Research Foundation (DFG), and co-directing the Heidelberg Collaboratory of Image Processing, funded by both industrial partners and the DFG. His research interests include mathematical modelling for computational image and pattern analysis.