Digital Signal Processing 23 (2013) 1844–1855

Contents lists available at SciVerse ScienceDirect

Digital Signal Processing

www.elsevier.com/locate/dsp

Bayesian 3D shape from silhouettes

Donghoon Kim, Jonathan Ruttle, Rozenn Dahyot *

School of Computer Science and Statistics, Trinity College Dublin, Ireland

ARTICLE INFO

Article history:
Available online 20 June 2013

Keywords:
3D reconstruction from multiple view images
Shape-from-silhouettes
Kernel density estimates
K-nearest neighbours
Principal component analysis

ABSTRACT

This paper introduces a smooth posterior density function for inferring shapes from silhouettes. Both the likelihood and the prior are modelled using kernel density functions and optimisation is performed using gradient ascent algorithms. Adding a prior allows for the recovery of concave areas of the shape that are usually lost when estimating the visual hull. This framework is also extended to use colour information when it is available in addition to the silhouettes. In these cases, the modelling not only allows for the shape to be recovered but also its colour information. Our new algorithms are assessed by reconstructing 2D shapes from 1D silhouettes and 3D faces from 2D silhouettes. Experimental results show that using the prior can assist in reconstructing concave areas and also illustrate the benefits of using colour information even when only small numbers of silhouettes are available.

1. Introduction

Three-dimensional reconstruction of an object that is seen by multiple image sensors has many applications such as 3D modelling [1,2] or video surveillance [3]. Shape from silhouettes methods infer the 3D shape of an object using a collection of its projected silhouette images captured from different points of view. The best possible reconstruction (called the visual hull [4]) can be computed using an infinite number of silhouettes captured from all viewpoints outside the convex hull of the object. Volume-based approaches focus on the volume of the visual hull [5–7,4] and this formulation can be re-expressed in a probabilistic framework to model uncertainty with a discrete cost function [8]. As an alternative, surface-based approaches aim to estimate a surface representation of the visual hull from the contours of the silhouette images [9–12], and Grauman et al. proposed a Bayesian framework for inferring a 3D surface using, as a shape representation, all contours of the silhouettes from multiple views [13]. However, surface-based approaches are less numerically stable than volumetric ones and are also more sensitive to segmentation error. Moreover, the visual hull does not capture concave regions of the 3D shape, and colour information can be used to palliate this limitation [14,15].

Volume-based approaches based on voxel occupancy rely on the optimisation of a discrete objective function [5–7,4]. The world volume is split into elementary blocks (voxels) and each block can project onto a pixel in the recorded silhouettes. Like the bin of a histogram, the block is incremented each time it projects onto the foreground part of a silhouette image. Such a representation corresponds to a histogram representation as an approximation of the probability density function of the spatial random variable $\mathbf{x}$ to be in the volume of the object. The quality of this reconstruction depends on the number of camera views, their viewpoints, the voxel resolution, and the complexity of the object. The discrete nature of the histogram makes the approach memory demanding. Moreover, optimisation methods (e.g. exhaustive search) of such discrete representations are limited and suboptimal compared to smooth modellings that can be optimised with gradient ascent methods. To alleviate this limitation, Kim et al. [16] recently proposed a smooth Kernel Density Estimate (KDE) as another approximation of the probability density function of the spatial random variable $\mathbf{x}$ to be in the volume of the object. For simplicity, their modelling considers a 3D object volume as seen by orthographic cameras. Ruttle et al. [17] extended this modelling to use standard pinhole cameras. Newton Raphson and Meanshift algorithms [16,17] can be used efficiently to search for the maxima of these KDEs and these are suitable for parallel programming using Graphics Processing Units (GPU) for instance [18,19]. These smooth KDEs [16,17] can be interpreted as likelihoods since they link the latent variable (i.e. the spatial position of the object $\mathbf{x}$) and the observations (silhouettes and camera parameters). However, without prior information about the object to be reconstructed, these modellings give an estimate of the visual hull and are therefore unable to reconstruct concave parts of the object.

To improve on the visual hull and recover concave regions, we propose here to extend this smooth modelling with KDEs by adding colour information in the likelihood and by adding prior information (Section 3). We assess our method experimentally (Section 4) and show that our approach accurately reconstructs the tested shapes. Accuracy, moreover, is enhanced when colour information is used. We first present our approach to model the

* Corresponding author.
E-mail address: Rozenn.Dahyot@scss.tcd.ie (R. Dahyot).

1051-2004/$ – see front matter
http://dx.doi.org/10.1016/j.dsp.2013.06.007

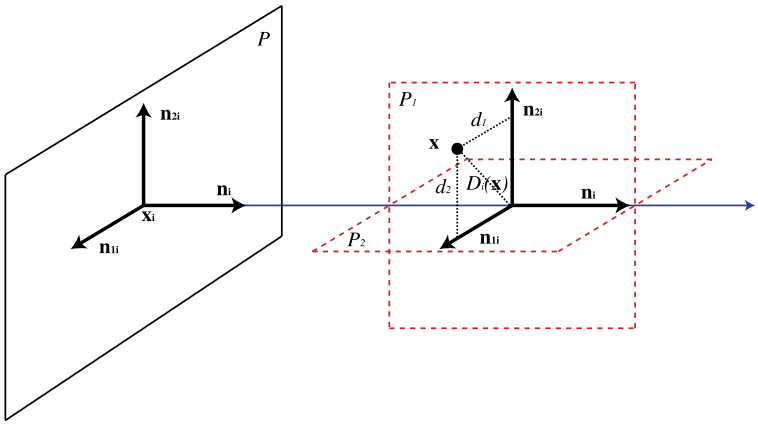


Fig. 1. A 3D ray modelled by the intersection of two orthogonal planes P_1 and P_2. $\mathbf{x} \in \mathbb{R}^3$ is a position in space and $\mathbf{x_i}$ is the spatial position of the pixel i in the image plane P. $D_i(\mathbf{x})$ is the shortest Euclidean distance from the 3D ray.

likelihood function in Section 2 using observations recorded by either orthographic or pinhole cameras [16,17,20]. This likelihood is completed by introducing a prior model using either K-Nearest Neighbours (KNN) [21] or Principal Component Analysis (PCA). Both approaches are encapsulated in a multiresolution framework to avoid local solutions. We assess the two algorithms respectively for 2D and 3D shape inference from respectively 1D and 2D silhouettes.

2. Modelling the likelihood

Our modelling for the likelihood originates from the following equation:

$$\lambda + F(\mathbf{x}, \Theta) = \epsilon \sim p_\epsilon(\epsilon) \tag{1}$$

where

- $\mathbf{x}$ is the latent spatial variable of interest: $\mathbf{x} \in \mathbb{R}^2$ is a spatial random variable in a plane (slice) when considering 2D shape inference from 1D silhouettes, and $\mathbf{x} \in \mathbb{R}^3$ is in the 3D space when performing 3D shape inference from 2D silhouettes.
- F is a given link function modelling the relation between the spatial position $\mathbf{x}$ and the information collected by the camera noted Θ (see Section 2.1).
- The observed random variable, Θ, is the projection of $\mathbf{x}$ in the image planes and many observations have been captured with the multiple cameras in the form of silhouette images. The camera parameters are all assumed to be known. The set $\{\Theta_i\}_{i=1,\dots,n}$ collects all observations from all pixels captured from different viewpoints.
- The random variable ϵ with distribution p_ϵ represents the noise that affects Θ. This distribution p_ϵ is the normal distribution with mean zero and variance h^2 in this paper.
- λ is an additive auxiliary variable in Eq. (1). Indeed Eq. (1) allows us to write:

  $$p_{\lambda|\Theta,\mathbf{x}}(\lambda|\Theta, \mathbf{x}) = p_\epsilon\big(\lambda + F(\mathbf{x}, \Theta)\big)$$

  The case of interest in this application is when $\lambda = 0$.

The joint density function of $\mathbf{x}$ and λ can be modelled by (assuming independence of Θ and $\mathbf{x}$):

$$\begin{aligned} &p_{\lambda,\mathbf{x}}(\lambda, \mathbf{x}) \\ &= \int p_{\lambda|\Theta,\mathbf{x}}(\lambda|\Theta, \mathbf{x}) p_{\mathbf{x}|\Theta}(\mathbf{x}|\Theta) p_\Theta(\Theta)\, d\Theta \quad \text{(Bayes)} \\ &= \int p_\epsilon\big(\lambda + F(\mathbf{x}, \Theta)\big) p_{\mathbf{x}|\Theta}(\mathbf{x}|\Theta) p_\Theta(\Theta)\, d\Theta \\ &\quad \big(p_\epsilon = p_{\lambda|\Theta,\mathbf{x}} \text{ see Eq. (1)}\big) \\ &= p_{\mathbf{x}}(\mathbf{x}) \int p_\epsilon\big(\lambda + F(\mathbf{x}, \Theta)\big) p_\Theta(\Theta)\, d\Theta \quad \text{(independence)} \\ &= p_{\mathbf{x}}(\mathbf{x}) \mathbb{E}_\Theta\big[p_\epsilon\big(\lambda + F(\mathbf{x}, \Theta)\big)\big] \quad \text{(expectation)} \end{aligned} \tag{2}$$

The joint density function of λ and Θ corresponds to the prior $p_{\mathbf{x}}(\mathbf{x})$ multiplied by an expectation that can be approximated by using the Strong Law of Large Numbers [22]:

$$\hat{p}_{\lambda,\mathbf{x}}(\lambda, \mathbf{x}) = \frac{1}{C} \underbrace{p_{\mathbf{x}}(\mathbf{x})}_{\text{prior}} \cdot \underbrace{\sum_{i=1}^{n} p_\epsilon\big(\lambda + F(\mathbf{x}, \Theta_i)\big)\, \pi_i}_{\text{when } \lambda=0,\ \overline{\text{lik}}(\mathbf{x})} \tag{3}$$

The observation Θ_i collected on pixel i has a weight π_i defined as $\pi_i = 0$ for a background pixel and $\pi_i = 1$ for a foreground pixel as defined by the binary silhouette images. The normalisation constant C is defined as $C = \sum_{i=1}^{n} \pi_i$. Note that this modelling also allows the handling of non-binary silhouettes if one chooses to use non-binary weights $\{\pi_i\}_{i=1,\dots,n}$. Inference about $\mathbf{x}$ can then be performed by exploring the likelihood $\overline{\text{lik}}(\mathbf{x})$ or the posterior $\hat{p}_{\lambda,\Theta}(\lambda = 0, \mathbf{x})$ when a prior is available. The term $\overline{\text{lik}}(\mathbf{x})$ can be understood as an average of likelihood functions computed with one observation at a time. More information about this inferential framework can be found in [23]. Next, we introduce explicit link functions F for two types of cameras.

2.1. Camera models

The definition of the link function F depends on the chosen camera model. In our framework, we consider two types of cameras: orthographic and pinhole. Orthographic camera models are not faithful representations of real cameras but they provide a connection with the Radon Transform (Section 2.1.1). Section 2.1.2 presents the link function for the pinhole camera model. Experimental results for 3D shape inference from silhouettes recorded by a pinhole camera using the cost function $\overline{\text{lik}}(\mathbf{x})$ are shown in Section 4.1.

2.1.1. Orthographic camera

The function F links the information recorded in the image with the 3D spatial position $\mathbf{x}$. Fig. 1 illustrates this relationship: each pixel i is characterised by a ray and all positions $\mathbf{x}$ onto this ray project exactly on this pixel. The further away position $\mathbf{x}$ is from the ray, the less influence the data recorded at pixel i has. To model this, we choose the Euclidean distance between the ray and $\mathbf{x}$ noted $D_i(\mathbf{x}) = F(\mathbf{x}, \Theta_i)$ where Θ_i corresponds to all known

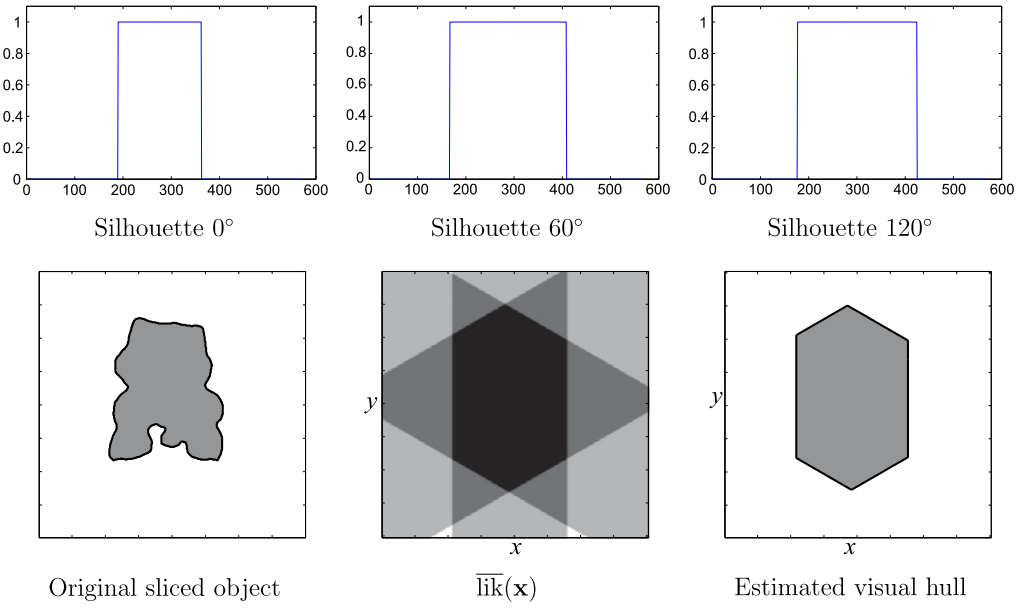


Fig. 2. One 2D slice of an object is shown in the bottom row, left. Three 1D silhouettes have been observed (top row) and used to estimate the likelihood (middle, bottom row: the blacker the colour of the map at $\mathbf{x} = (x, y)$, the higher the value of $\overline{\text{lik}}(\mathbf{x})$). The estimated visual hull (shown bottom row, right) can be then extracted (Section 2.3). The contour of the estimated visual hull is used as the initial estimate of our algorithms (Section 3). Note that as more silhouettes are captured, the estimate converges towards the visual hull (e.g. Fig. 3 below shows $\overline{\text{lik}}(\mathbf{x})$ computed with 36 silhouettes).

parameters associated with pixel i (normal vectors are noted as $\mathbf{n}_{1i}$ and $\mathbf{n}_{2i}$, and spatial position of pixel i is noted as $\mathbf{x}_i$):

$$D_i(\mathbf{x})^2 = d_1^2 + d_2^2 = \left(\mathbf{n}_{1i}^T(\mathbf{x} - \mathbf{x}_i)\right)^2 + \left(\mathbf{n}_{2i}^T(\mathbf{x} - \mathbf{x}_i)\right)^2 \quad (4)$$

By setting one of the coordinates of $\mathbf{x}$ to be equal to a constant, the cost function $\overline{\text{lik}}(\mathbf{x})$ can then be visualised in a 2D slice of the object. As an illustration, Fig. 2 shows this cost function with $\mathbf{x} \in \mathbb{R}^2$ in a slice where the 1D silhouettes correspond to a single line in the binary 2D silhouette image. Pintavirooj et al. have proposed to solve shape from silhouettes using the inverse Radon Transform for reconstruction in a stack of 2D slices [24] and Kim et al. have shown how inference with the smooth cost function $\overline{\text{lik}}(\mathbf{x})$ in 2D slices outperforms the discrete inverse Radon Transform approach [16].

2.1.2. Pinhole camera

For the pinhole camera, the function $F(\mathbf{x} = (x, y, z), \Theta = (\theta_1, \theta_2, P))$ is defined as [17]:

$$F(\mathbf{x}, \Theta) = \begin{pmatrix} F_1(\mathbf{x}, \Theta) = \theta_1 - \frac{x\,P_{11} + y\,P_{12} + z\,P_{13} + P_{14}}{x\,P_{31} + y\,P_{32} + z\,P_{33} + P_{34}} \\ F_2(\mathbf{x}, \Theta) = \theta_2 - \frac{x\,P_{21} + y\,P_{22} + z\,P_{23} + P_{24}}{x\,P_{31} + y\,P_{32} + z\,P_{33} + P_{34}} \end{pmatrix} \quad (5)$$

where P holds the camera matrix parameters that are known. The observation Θ_i for pixel i corresponds to its pixel position in the image $(\theta_{1i}, \theta_{2i})$ and its camera parameters P_i. Reconstruction with the likelihood with a pinhole camera has been assessed on the Middlebury dataset [25,17] and additional results are presented in Section 4.1.

2.2. Extension to colour

The use of colour in volumetric reconstruction methods is also well studied in the context of shape carving and of reconstructing the photo hull [14,15]. Indeed, using photo-consistency from multiple view points can also help in recovering concavities. Our modelling is extended to use both colour and silhouette information for the inference of a coloured 3D shape. In order to take colour into account, RGB values of the pixels are converted to chromaticity values since chromaticity red and green are more invariant to lighting conditions [26,27]. The conversion equation is as follows:

$$r = \frac{R}{R + G + B}, \qquad g = \frac{G}{R + G + B} \quad (6)$$

Two additional Gaussian probabilities of chromaticity red r and green g are added to the likelihood of the KDEs. The colour KDE is then generalised as:

$$\hat{p}_{\lambda \mathbf{x} rg}(\lambda = 0, \mathbf{x}, r, g) \propto p_{\mathbf{x} rg}(\mathbf{x}, r, g) \times \underbrace{\sum_{i=1}^{n} \exp\left(\frac{-F(\mathbf{x}, \Theta_i)^2}{2h^2} - \frac{(r - r_i)^2}{2h_r^2} - \frac{(g - g_i)^2}{2h_g^2}\right) \pi_i}_{\overline{\text{lik}}(\mathbf{x}, r, g)} \quad (7)$$

where (h, h_r, h_g) are the bandwidths of the Gaussian kernels for the spatial and colour domains. Such modelling allows for not only the recovery of shapes but also of shapes' surface colours (photo hull).

2.3. Optimisation

The cost function $\overline{\text{lik}}(\mathbf{x})$ is a KDE computed using n observations. As n becomes larger, i.e. when more images are collected, the computation of $\overline{\text{lik}}(\mathbf{x})$ at a spatial position $\mathbf{x}$ becomes more intensive. In practise, we consider only the kernels with the observations (pixels) in the vicinity of the projection of $\mathbf{x}$ in each camera view. This reduces the number of computations needed to evaluate $\overline{\text{lik}}(\mathbf{x})$ at $\mathbf{x}$. The contour of the convex hull of the object can then be recovered using $\overline{\text{lik}}(\mathbf{x})$ computed on a grid spanning the 2D slice [17]. Alternatively, gradient ascent techniques can also be used to find this convex hull [16,17]. In particular, the Meanshift algorithm has been used for optimising both the likelihood $\overline{\text{lik}}(\mathbf{x})$ (and $\overline{\text{lik}}(\mathbf{x}, r, g)$) and the posteriors when using an orthographic camera model. Newton Raphson algorithms have been used when using a pinhole camera [17,28].

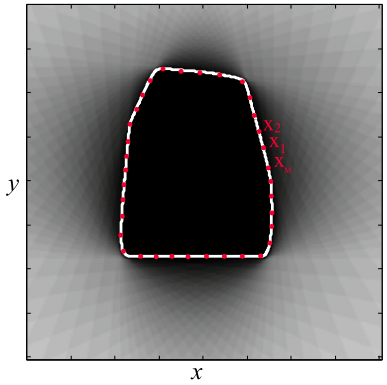


Fig. 3. View of $\overline{\text{lik}}(\mathbf{x} = (x, y))$ computed for the ALOI object (see Fig. 2) with 36 silhouettes. The contour is extracted and the feature vector collects all spatial positions $\mathbf{x}_1, \ldots, \mathbf{x}_M$ around the contour and also the colour when available (Section 3.1). At the start of the algorithms, this contour extracted from the likelihood is the first guess $\hat{\mathbf{X}}^{(0)}$.

3. Modelling of the posterior

Two modellings for the prior are proposed to complement the likelihood, one using KNN (Section 3.2.1) and one using PCA (Section 3.2.2). This prior is currently designed for shapes in 2D; 3D reconstruction is consequently performed using a stack of 2D slices. First, we start by describing our shape representation (Section 3.1) and our prior (Section 3.2). Resulting posteriors are presented in Section 3.3 and the multiresolution approach in Section 3.4. Resulting algorithms for inference of shape are summarised in Section 3.5.

3.1. Shape description

The shape is described by a sequence of connected points. The points are chosen uniformly along the contour in an anti-clockwise direction (see Fig. 3). Note the sequence of points is normalised by subtracting the mean of the points coordinates and by dividing by their respective variance. This is a standard pre-processing step for obtaining a representation that is invariant to translation and scale. Our shape descriptor contains not only the ordered list of 2D points $\{\mathbf{x}_i = (x_i, y_i)\}_{i=1,\ldots,M}$ but also its local angles $\{\alpha_i\}_{i=1,\ldots,M}$: α_i is the angle between the vectors $\mathbf{x}_i - \mathbf{x}_{i+1}$ and $\mathbf{x}_i - \mathbf{x}_{i-1}$. We define the function f as:

$$f(\mathbf{X}) = \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \vdots \\ \alpha_M \end{bmatrix} \quad \text{with} \quad \mathbf{X} = \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \mathbf{x}_3 \\ \vdots \\ \mathbf{x}_M \end{bmatrix} \tag{8}$$

where M is the number of sampled points to describe the shape. Note that $\mathbf{X}$ and $f(\mathbf{X})$ are not invariant to rotation, i.e. choosing a different starting point $\mathbf{x}_1 = (x_1, y_1)$ on the shape will lead to other vectors $\mathbf{X}'$ and $f(\mathbf{X}')$ that will be cyclic permutations of $\mathbf{X}$ and $f(\mathbf{X})$. Note also that the representation $f(\mathbf{X})$ is, however, invariant to scale changes on $\mathbf{X}$.

Colour information can also be added when available in the shape descriptor such that the chrominance information (Eq. (6)) for each location $\mathbf{x}_i$ on the contour also appears in the descriptor $\mathbf{X}$. The spatial coordinates (x, y) and chromaticity values (r, g) are used to create the feature vector and, therefore, the dimension of the feature vector is $4 \times M$, where M is the number of points on the contour.

3.2. Shape prior modelling

Having a training database of N shape exemplars, a standard approach is to compress the information in the training database and extract a small basis of functions to accurately reconstruct any shape $\mathbf{X}$ with a small error. We propose two bases of functions, KNN (Section 3.2.1) and PCA (Section 3.2.2), that can be used for reconstruction. The exemplars $\{\mathbf{X}_j^e\}_{j=1,\ldots,N}$ used for training are all normalised as described in Section 3.1 to remove the effects of scale and translation.

3.2.1. Shape prior modelling using KNN

The distance metric between shapes $\mathbf{X}$ and $\mathbf{Y}$ is defined as follows:

$$d(\mathbf{X}, \mathbf{Y}) = \sum_{i=0}^{M} \left| \alpha_i^{\mathbf{X}} - \alpha_i^{\mathbf{Y}} \right| \tag{9}$$

This metric is an absolute distance between $f(\mathbf{X})$ and $f(\mathbf{Y})$ and is used to find the nearest neighbours. We define our basis of functions $\{\mathbf{U}_k\}_{k=1,\ldots,K}$ by selecting the K exemplars of the training database that will be at the shortest distance of a shape $\mathbf{X}$. To be insensitive to rotation, we also consider all cyclic permutations of the exemplars. For instance, considering the first exemplar $\mathbf{X}_1^e$, we find its cyclic permutation m (noted $\mathbf{X}_1^{e(m)}$) to have the minimum distance $d(\mathbf{X}, \mathbf{X}_1^{e(m)})$ defined by:

$$\hat{m}_1 = \arg\min_m \left\{ d\left(\mathbf{X}, \mathbf{X}_1^{e(m)}\right) \right\} \tag{10}$$

Having computed all best distances between $\mathbf{X}$ and the N exemplars:

$$\left\{ d\left(\mathbf{X}, \mathbf{X}_1^{e(\hat{m}_1)}\right), d\left(\mathbf{X}, \mathbf{X}_2^{e(\hat{m}_2)}\right), \ldots, d\left(\mathbf{X}, \mathbf{X}_N^{e(\hat{m}_N)}\right) \right\} \tag{11}$$

the K exemplars with the shortest distances are then selected. The reconstruction $\mathbf{X}_U$ is defined on the basis of these selected K-nearest neighbours (noted $\mathbf{U}_k$):

$$\mathbf{X}_U = \sum_{k=1}^{K} \omega_k \, \mathbf{U}_k \tag{12}$$

The weights $\{\omega_k\}_{k=1,\ldots,K}$ are calculated as follows:

$$\omega_k = \frac{1}{(K-1)} \left(1 - \frac{d_k}{d_{sum}} \right)$$

with

$$d_{sum} = \sum_{k=1}^{K} d(\mathbf{X}, \mathbf{U}_k) \quad \text{and} \quad d_k = d(\mathbf{X}, \mathbf{U}_k) \tag{13}$$

Note that the weights sum to 1, $\sum_{k=1}^{K} \omega_k = 1$. The reconstruction $\mathbf{X}_U$ approximates the observed shape $\mathbf{X}$ but at a normalised scale since the exemplars in the training database are all normalised.

3.2.2. Shape prior modelling using PCA

The PCA-based representation has been widely used to model shapes such as faces [29,30] and also in Active Appearance/Shape Model [31–34]. PCA allows a shape to be approximated by a linear combination of eigenvectors of the covariance matrix computed using the exemplars. To remove the effect of rotation, we select the best cyclic permutation $\mathbf{X}_j^{e(\hat{m}_j)}$ of the exemplar $\mathbf{X}_j^e$ first for each exemplar. The covariance matrix is computed with the N exemplars $\{\mathbf{X}_j^{e(\hat{m}_j)}\}_{j=1,\ldots,N}$ and its first K eigenvectors associated with the K highest eigenvalues are computed with singular value decomposition [35]. The reconstruction $\mathbf{X}_U$ of a normalised shape $\mathbf{X}$

is computed as the linear combination of the mean shape $\boldsymbol{\mu}$ and the K eigenvectors:

$$\mathbf{X}_U = \boldsymbol{\mu} + \sum_{k=1}^{K} \omega_k \mathbf{U}_k \quad \text{with } \omega_k = \langle \mathbf{X} - \boldsymbol{\mu} | \mathbf{U}_k \rangle \tag{14}$$

where the mean is $\boldsymbol{\mu} = \frac{1}{N}\sum_{j=1}^{N} \mathbf{X}_j^{e(\hat{m}_j)}$ and $\mathbf{U}_k$ is the eigenvector associated with the kth highest eigenvalue. The disadvantage of the PCA-based method is that the normalisation step to remove the effects of translation and scaling is required for the current observation $\mathbf{X}$ to become as similar as possible to the training database $\{\mathbf{X}_i^{e(\hat{m}_i)}\}_{i=1,\dots,N}$. Note, that using KNN with our shape descriptor does not require these normalisation steps to be taken between the observation $\mathbf{X}$ and training database.

3.3. Posterior and inference

Having a current guess of the contour noted $\hat{\mathbf{X}}^{(t)}$, we can compute the reconstruction $\mathbf{X}_U^{(t)} = [\mathbf{x}_{U_1}^{(t)}, \dots, \mathbf{x}_{U_M}^{(t)}]$. We model a prior using $\mathbf{X}_U^{(t)}$ to allow for the estimation of the refined shape at $(t+1)$. Each of the M points of $\mathbf{X}^{(t+1)} = [\mathbf{x}_1^{(t+1)}, \dots, \mathbf{x}_M^{(t+1)}]$ is updated individually. Let us consider the first point $\mathbf{x}_1^{(t+1)}$. The likelihood is modelled using the KDE (Eq. (3)) and the prior for $\mathbf{x}_1^{(t+1)}$ is modelled given $\mathbf{X}_U^{(t)}$ and $\hat{\mathbf{X}}^{(t)}$:

$$\text{post}\big(\mathbf{x}_1^{(t+1)}\big) \propto \overline{\text{lik}}\big(\mathbf{x}_1^{(t+1)}\big) \times \text{prior}\big(\mathbf{x}_1^{(t+1)} | \mathbf{X}_U^{(t)}, \hat{\mathbf{X}}^{(t)}\big) \tag{15}$$

The reconstruction $\mathbf{X}_U^{(t)}$ is converted into $M-1$ unit vectors $\{\mathbf{n}_m^{(t)}\}_{m=2,\dots,M}$ such that $\mathbf{n}_m^{(t)}$ is orthogonal to the line defined by $(\mathbf{x}_{U_1}^{(t)}, \mathbf{x}_{U_m}^{(t)})$. We assume that the update $\mathbf{x}_1^{(t+1)}$ is in the neighbourhood of the line orthogonal to $\mathbf{n}_m^{(t)}$ and going through the point $\hat{\mathbf{x}}_m^{(t)}$. This can be translated into the following equation:

$$\mathbf{n}_m^{(t)T}\big(\mathbf{x}_1^{(t+1)} - \hat{\mathbf{x}}_m^{(t)}\big) = \epsilon_p \tag{16}$$

where $\epsilon_p \sim \mathcal{N}(0, h_p^2)$ is the error with normal distribution (mean 0, variance h_p^2). In a similar fashion as the likelihood, the prior is modelled using a KDE:

$$\text{prior}\big(\mathbf{x}_1^{(t+1)} \big| \mathbf{X}_U^{(t)}, \hat{\mathbf{X}}^{(t)}\big) \propto \sum_{m=2}^{M} \exp\left(-\frac{(\vec{n}_m^T(\mathbf{x}_1^{(t+1)} - \hat{\mathbf{x}}_m^{(t)}))^2}{2h_p^2}\right) \tag{17}$$

We use only slopes from the reconstruction and therefore this method is invariant to scale difference between the shape $\hat{\mathbf{X}}^{(t)}$ and the normalised reconstruction $\mathbf{X}_U^{(t)}$. Since both the likelihood and the prior are KDEs, the posterior distribution is also a KDE and a gradient ascent algorithm is used here to maximise the posterior:

$$\hat{\mathbf{x}}_1^{(t+1)} = \arg\max_{\mathbf{x}_1^{(t+1)}} \big\{\text{post}\big(\mathbf{x}_1^{(t+1)}\big)\big\} \tag{18}$$

This is repeated for each point in the contour such that the estimated update is computed:

$$\hat{\mathbf{X}}^{(t+1)} = \big[\hat{\mathbf{x}}_1^{(t+1)}, \dots, \hat{\mathbf{x}}_M^{(t+1)}\big]$$

The shape of the initial guess $\hat{\mathbf{X}}^{(0)}$ is the result of the estimation using only the likelihood [16] (Fig. 3).

3.4. A coarse-to-fine strategy

In order to converge iteratively towards a good solution even if the starting guess $\hat{\mathbf{X}}^{(0)}$ (e.g. the convex approximation to the shape) is far from it, we need to be careful when modelling the prior. Indeed at the start, the reconstruction $\mathbf{X}_U^{(0)}$ may not be very accurate. To avoid this problem, we construct a Gaussian shape stack whose concept is introduced in Lefebvre and Hoppe [36]. The Gaussian stack is constructed by smoothing the exemplar shapes in the prior set using increasing bandwidths (noted $h_e^{(t)}$) without downsampling the shapes as is usually done in Gaussian pyramids. This stack is computed using the convolution with a Gaussian (with bandwidth $h_e^{(t)}$) on all exemplars in the training database from large to small bandwidths as a smoothing factor. We note $\mathcal{S}_{prior}^{h_e(t)}$ to be the set of exemplars smoothed with a Gaussian of bandwidth $h_e^{(t)}$. The bandwidth $h_e^{(t)}$ decreases at each iteration of the algorithm as follows:

$$h_e^{(t)} = \alpha^t h_{\max} \quad \text{until} \quad h_e \leqslant h_{\min} \quad \text{with } \alpha = 0.9 \tag{19}$$

where $h_{\max} = 13$, $h_{\min} = 1$ and $h_{\max}$ is selected experimentally. This procedure allows us to achieve a coarse-to-fine strategy in modelling the prior. Fig. 4 shows how an exemplar shape evolves from a smooth convex shape to a more structured one as the bandwidth h_e decreases. The reconstruction at time t, $\mathbf{X}_U^{(t)}$, that is approximated from the selected exemplars in the training database is then iteratively refined to get more accurate shape estimates.

3.5. Algorithms

The estimation procedure using KNN is summarised in Algorithm 1:

Algorithm 1 Shape from Silhouettes using KNN prior

Computation of an initial guess $\hat{\mathbf{X}}^{(0)}$ of the shape at time $t=0$ with the likelihood [16]
Init $h_e(0) = h_{\max} = 13$
repeat
  Select the K nearest exemplars of $\hat{\mathbf{X}}^{(t)}$ in $\mathcal{S}_{prior}^{h_e(t)}$ and compute $\mathbf{X}_U^{(t)}$ (Eq. (12))
  for $i = 1 \to M$ **do**
    Model the prior for $\mathbf{x}_i^{(t+1)}$ (Eq. (17))
    $\hat{\mathbf{x}}_i^{(t+1)} = \arg\max_{\mathbf{x}_i^{(t+1)}} \{\text{post}(\mathbf{x}_i^{(t+1)})\}$
  end for
  $t \leftarrow t+1$
  $h_e(t) = \alpha^t\ h_{\max}$ with $\alpha = 0.9$
until $h_e(t) \leqslant h_{\min} = 1$

The estimation procedure using PCA is summarised in Algorithm 2:

Algorithm 2 Shape from Silhouettes using PCA prior

Computation of an initial guess $\hat{\mathbf{X}}^{(0)}$ of the shape at time $t=0$ with the likelihood [16]
Init $h_e(0) = h_{\max} = 13$
repeat
  Compute PCA using exemplars in $\mathcal{S}_{prior}^{h_e(t)}$ and select the K eigenvectors associated with the highest eigenvalues. Normalise $\hat{\mathbf{X}}^{(t)}$ and compute $\mathbf{X}_U^{(t)}$ (Eq. (14)).
  for $i = 1 \to M$ **do**
    Model the prior for $\mathbf{x}_i^{(t+1)}$ (Eq. (17))
    $\hat{\mathbf{x}}_i^{(t+1)} = \arg\max_{\mathbf{x}_i^{(t+1)}} \{\text{post}(\mathbf{x}_i^{(t+1)})\}$
  end for
  $t \leftarrow t+1$
  $h_e(t) = \alpha^t\ h_{\max}$ with $\alpha = 0.9$
until $h_e(t) \leqslant h_{\min} = 1$

The proposed prior is updated iteratively so that concavity information can be introduced progressively using the Gaussian stack. The prior is also refined at each step by choosing the nearest neighbours of the current estimate (KNN) and by recalculating the eigenvectors (PCA). Both our approaches refine the selection of these components iteratively during the estimation. This strategy differs from standard approaches where the reconstruction is computed as a linear combination of K fixed pre-selected components.

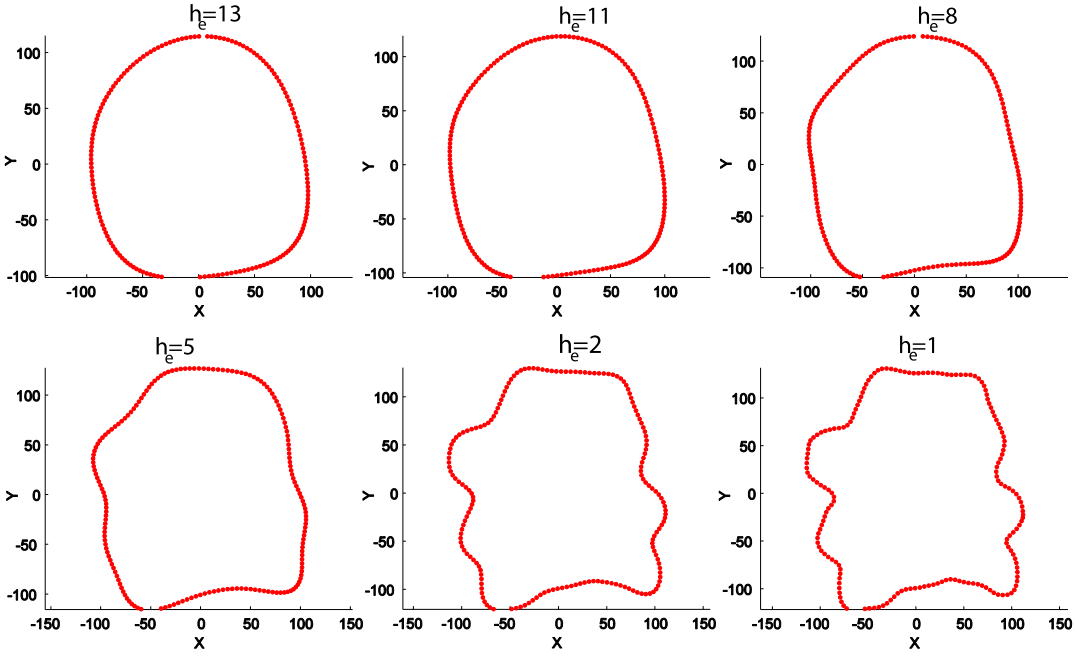


Fig. 4. Multiresolution approach: variations of one of the exemplars in the training database w.r.t. h_e.

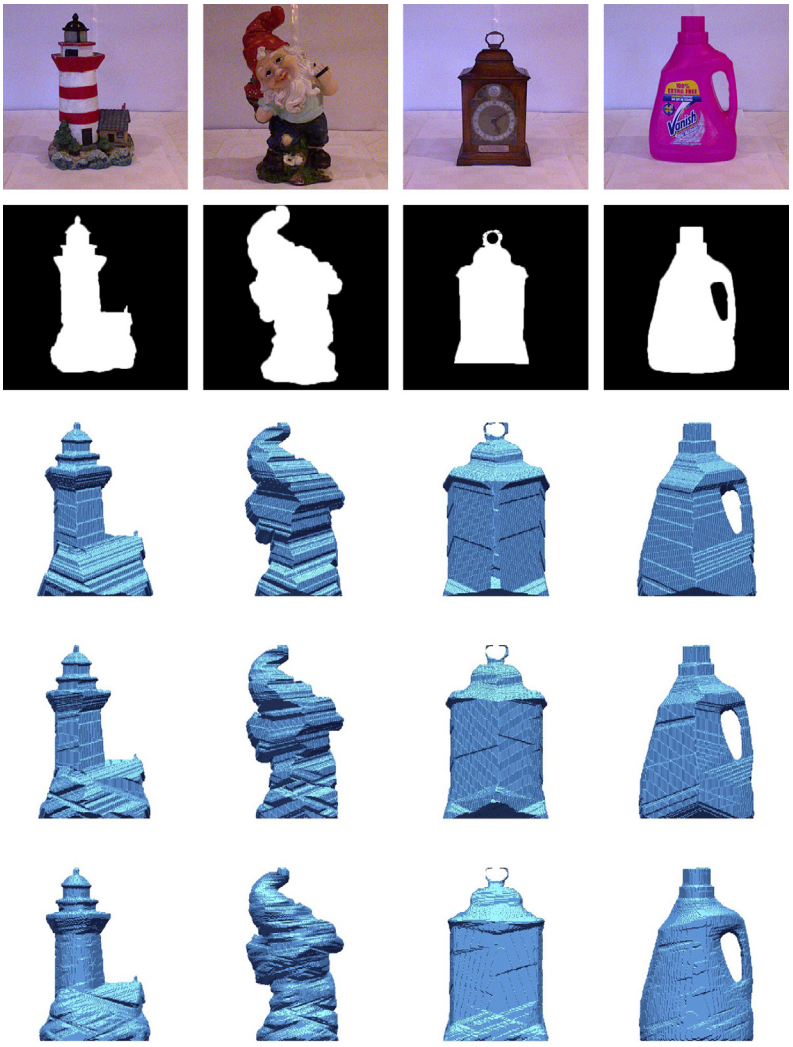


Fig. 5. Shape from silhouette reconstruction using the likelihood $\overline{\text{lik}}(\mathbf{x})$ (modelled with $\mathbf{x} \in \mathbb{R}^3$). From top to bottom: original objects, one silhouette, reconstruction from respectively 3, 6 and 36 camera views (i.e. silhouettes).

4. Experimental results

Section 4.1 shows the 3D reconstruction using only the likelihood with real images captured on a turning table. Section 4.2 assesses our methods for the reconstruction of 2D shapes from 1D silhouettes. Section 4.3 extends our approaches to 3D shape inference from 2D silhouettes.

4.1. 3D reconstruction using $\overline{lik}(\mathbf{x})$

Fig. 5 shows the 3D reconstructions computed using the likelihood $\overline{\text{lik}}(\mathbf{x})$ (with a pinhole camera model) on real objects captured with a turning table. The inference is directly performed in the 3D space ($\mathbf{x} \in \mathbb{R}^3$). Note how the concavity is not recovered: as more camera views are available, the reconstruction converges towards the visual hull. This reconstruction using the likelihood has been assessed and compared with the Middlebury dataset [25,17] using silhouettes. Inference with our gradient ascent algorithms has shown to be advantageous in terms of both memory requirements and computation time [17,28].

4.2. 2D shape reconstruction using KNN

This experiment assesses our approach for 2D shape reconstruction from 1D silhouettes using a projective camera model.

4.2.1. Training and test databases

The 2D shapes that model the prior are the contours of 6 objects taken from the ALOI database [37]. Each object class has seven images recorded from different viewing angles $[0°, 15°, 30°, 45°, 60°, 75°, 90°]$ which are divided into the test database $\mathcal{S}_{test}$ (angles $[15°, 45°, 75°]$) and the training database $\mathcal{S}_{prior}$ (angles $[0°, 30°, 60°, 90°]$) (see Fig. 6). The training database $\mathcal{S}_{prior}$ is used to approximate the prior for the shape with KNN. The total number of exemplars in the training set $\mathcal{S}_{prior}$ is $N = 6 \times 4 = 24$. The exemplar $\mathbf{X}^e$ is sampled into $M = 360$ points to represent its contour.

4.2.2. Observations

The observed silhouettes correspond to 1D binary signals: the contours in $\mathcal{S}_{test}$ are back-projected using orthographic projection in different directions. These projections are computed using the Radon Transform that are then thresholded to give binary silhouettes. These binary 1D silhouettes are used to compare the reconstructions inferred using the likelihood and the ones inferred using the posterior. Colour information on the foreground is also used to design one of the posteriors and we assess next its benefits compared to using only the silhouettes.

4.2.3. Experiments

In this section we compare the following 2D reconstructions:

- $\hat{\mathbf{X}}_1$ inferred using the likelihood computed with the silhouettes as observations,
- $\hat{\mathbf{X}}_2$ inferred using the posterior (KNN $K = 2$) computed with the silhouettes as observations,
- $\hat{\mathbf{X}}_3$ inferred using the posterior (KNN $K = 2$) computed with the silhouettes and the foreground colour as observations.

Having the ground truth shape $\mathbf{O}$, the Euclidean distance $d_i = \|\hat{\mathbf{X}}_i - \mathbf{O}\|$, $\forall i = 1, 2, 3$ is computed to assess the reconstructions. The distances are computed for all shapes in the test set $\mathcal{S}_{test}$ and their averages over all the shapes, $\bar{d}_i$, are computed with their standard errors. Fig. 7 shows $\bar{d}_1$, $\bar{d}_2$ and $\bar{d}_3$ w.r.t. the number of views (number of projections or 1D silhouettes available). The distance $\bar{d}_1$ can only decrease up to a point where the visual hull is recovered since only the likelihood is used. The distance $\bar{d}_2$ is lower than $\bar{d}_1$ because the posterior allows for the recovery of concave regions of the shape. This is only the case, however, when a sufficient number of views are available (superior to 7) and we note that the standard errors of $\bar{d}_2$ are quite large when very few cameras are used. Indeed, if the shape cannot be well discriminated from different viewing angles using only silhouette information, it becomes hard to choose the optimal exemplars (the K neighbours) to compute the prior. Also, there are sometimes problems in finding the best cyclic permutation of an exemplar $\mathbf{X}^{e(\hat{m})}$ which can be misleading when trying to create the prior for the shape. However, we note that the performance of the reconstruction $\hat{\mathbf{X}}_2$ is better than $\hat{\mathbf{X}}_1$ computed with the likelihood when

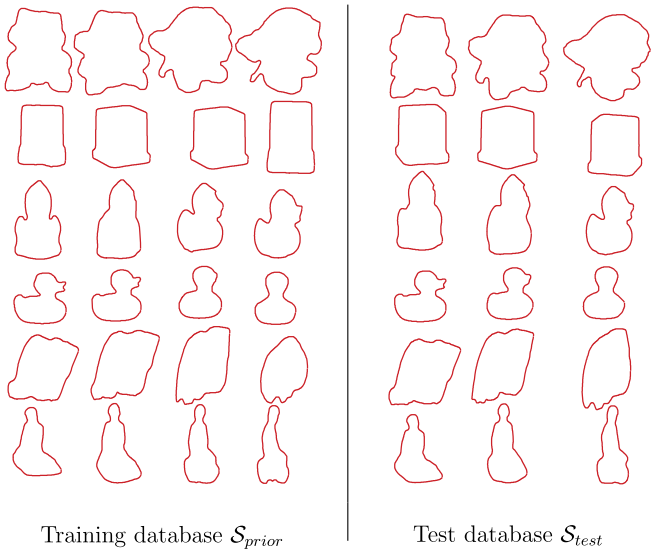


Fig. 6. Databases of 2D shapes at the highest resolution used for training and testing our algorithms. All cyclic permutations of these exemplars are also taken into account to allow the reconstruction process to be insensitive to rotations. When colour information is used, the colour on the contours in the original images of the ALOI database [37] is used.

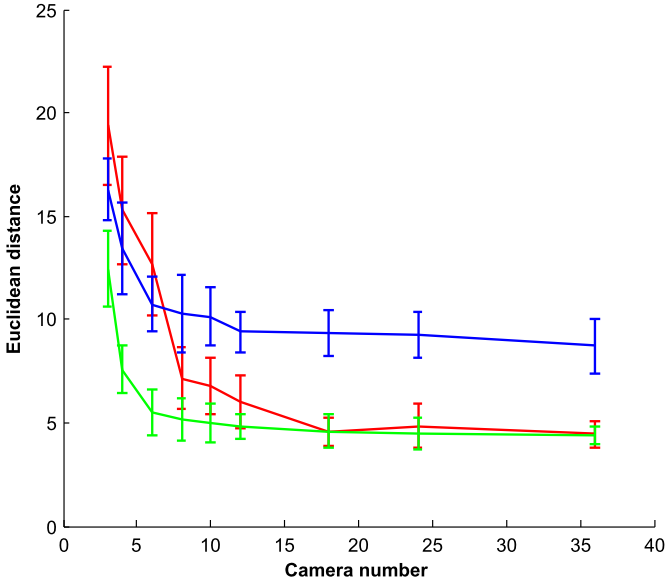
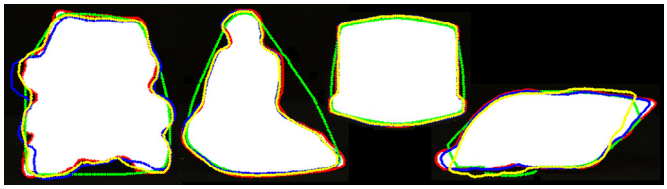
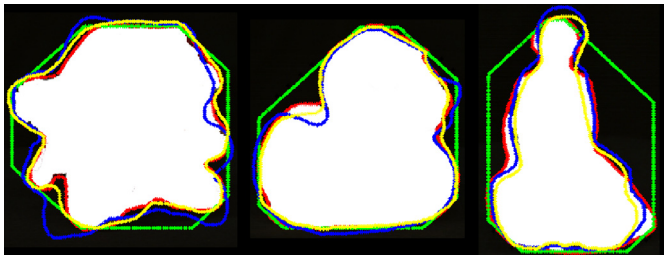


Fig. 7. Euclidean distance plot with standard error w.r.t. the number of camera views: $\bar{d}_1$ (blue), $\bar{d}_2$ (red) and $\bar{d}_3$ (green). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)



(a) Results from 36 camera views



(b) Results from 3 camera views

Fig. 8. Reconstructions: groundtruth $\mathbf{O}$ (red), $\hat{\mathbf{X}}_1$ (green), $\hat{\mathbf{X}}_2$ (blue) and $\hat{\mathbf{X}}_3$ (yellow). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

more than 7 cameras are used. When adding colour information, $\bar{d}_3$ is the smallest overall indicating that $\hat{\mathbf{X}}_3$ is the best reconstruction regardless of the number of cameras used.

The 2D reconstructions are shown in Fig. 8. In general, concavities are well recovered using the posteriors compared to the likelihood unless there are not enough clues (in the observations) to guide the selection of the best exemplars. Overall the results show that the posteriors are able to recover the concave parts of the object and perform better than the likelihood.

Fig. 9 shows the reconstructions $\hat{\mathbf{X}}_3$ at different resolutions of our algorithm. Concavity is introduced iteratively by decreasing the smoothing parameter h_e. We can see that the reconstruction is very close to the ground truth (i.e. corresponding smoothed exemplar) at each resolution level.

4.3. 3D face reconstruction using PCA

In this section, the prior is modelled using PCA as the eigenvectors are known to be an excellent basis to represent faces, both for 2D images [38] and 3D scans [39].

4.3.1. Database

We used the 3D Basel face model [39] which was created using 200 registered faces acquired with a structured light scanner (Fig. 10). Synthetic faces can be generated from random model coefficients as proposed by Paysan et al. [39]. For our experiments, a total of 44 faces were created from the 3D Basel face model: 9 faces are used for the test set $\mathcal{S}_{test}$ and 35 faces ($N = 35$) are for the training set $\mathcal{S}_{prior}$. To create silhouettes in multiple views, the 3D faces are projected in several directions using an orthographic projection. The 3D faces have been split into 70 horizontal slices. A 3D reconstruction is then computed by stacking the estimated horizontal 2D reconstructions along the Z-axis. The faces only correspond to a truncated head (see Fig. 11) where the back of the head is ignored. This truncation makes the alignment in rotation easier in our algorithm (i.e. contrary to Section 4.2, there is no real ambiguity in the projections and therefore the prior is not prone to error in finding the best cyclical permutation of the exemplars). The novelty in this experiment is to adapt the 2D shape inference scheme proposed in this paper for 3D shape inference. In particular we need the selected prior for each 2D stack.

4.3.2. Modelling the 3D prior

The shape descriptor is redefined as follows for the PCA priors:

$$\mathbf{X} = [\mathbf{S}_1; \mathbf{S}_2; \cdots; \mathbf{S}_s] \tag{20}$$

where $\mathbf{S}_i$ is the contour described by $M = 360$ points in the 2D ith slice and s is the total number of the 2D slices (here $s = 70$). The order of the 2D slices to create the feature vector is from top to bottom and the sequence of the points for the slice is in an anti-clockwise direction (Fig. 11). The 3D prior $\mathbf{X}_U$ in Algorithm 2 for the 3D faces is computed at each resolution by finding the best rotations of all the 3D exemplars in $\mathcal{S}_{prior}$ to align them with the current 3D reconstruction $\hat{\mathbf{X}}$. Then PCA is computed for each slice using the aligned exemplars and only $K = 3$ eigenvectors are used to update the prior in each slice.

4.3.3. 3D reconstructions of faces

In this section, we compare the following 3D reconstructions:

- $\hat{\mathbf{X}}_4$ inferred using the posterior (PCA $K = 3$) computed with the silhouettes as observations,
- $\hat{\mathbf{X}}_5$ inferred using the posterior (PCA $K = 3$) computed with the silhouettes and the foreground colour as observations.

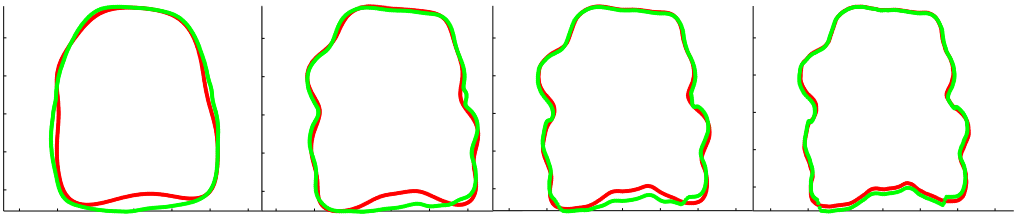


Fig. 9. From left to right: coarse-to-fine evolution of the reconstructions $\hat{\mathbf{X}}_3$ (red) with the ground truth (green) at the same resolution level. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Fig. 10. Examples in the Basel face model [39].

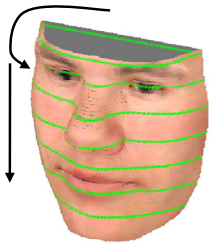


Fig. 11. 3D face decomposed into 2D oriented slices.

36 orthographic binary projections (2D silhouettes) were used in this experiment in addition to foreground colour information for computing $\hat{\mathbf{X}}_5$. Fig. 12 presents some 3D reconstruction results with their error surfaces. The error surfaces show the distances between the points of the reconstructions and the closest polygon in the mesh of the ground truth. The average height of the 3D heads in the database is 150 mm and the error ranges from 0 mm to 10 mm in the error surfaces.

Table 1
Means and standard deviations (in mm) of the absolute error between the ground truth GT and the reconstructions, $\hat{\mathbf{X}}_4$ and $\hat{\mathbf{X}}_5$, and also the PCA mean $\boldsymbol{\mu}$ (see Fig. 12 for the error surfaces computed for faces (1)–(3)).

	$\hat{\mathbf{X}}_4$	$\hat{\mathbf{X}}_5$	$\boldsymbol{\mu}$
(1) mean	2.258	2.1836	3.329
(1) std	1.7246	1.7108	4.2664
(2) mean	2.3504	2.2651	3.1239
(2) std	1.6866	1.6555	3.9391
(3) mean	2.4344	2.2633	2.5122
(3) std	1.825	1.802	3.4992

Table 1 shows the means and standard deviations of the errors (shown as error surfaces in Fig. 12) when comparing the reconstructions with the ground truth. We have also computed the mean and standard deviation of the errors when comparing the average face $\boldsymbol{\mu}$ given by PCA with the ground truth. We note that using $K = 3$ principal components, both reconstructions $\hat{\mathbf{X}}_4$ and $\hat{\mathbf{X}}_5$

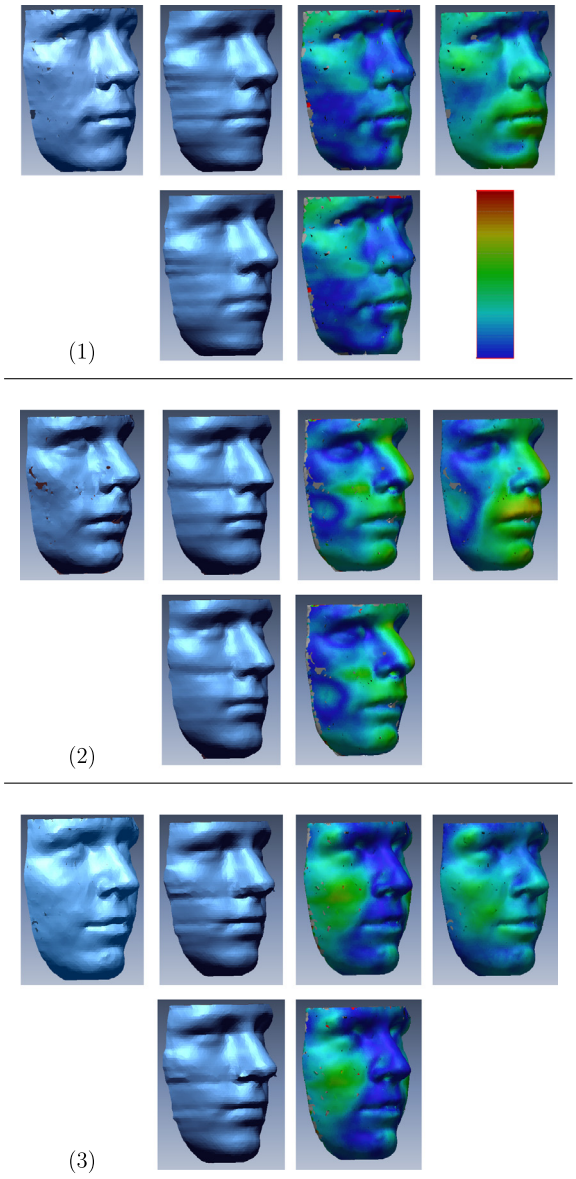


Fig. 12. 3D face reconstruction: groundtruth (GT, left), the estimates (middle) ($\hat{\mathbf{X}}_4$ is on top of $\hat{\mathbf{X}}_5$ in each case), the corresponding error surfaces $|\hat{\mathbf{X}}_4 - GT|$ and $|\hat{\mathbf{X}}_5 - GT|$ compared with the error with the PCA mean $|GT - \mu|$ (right). The colour scale for all the error surfaces is also shown ranging from 0 mm (blue) to 10 mm (red).

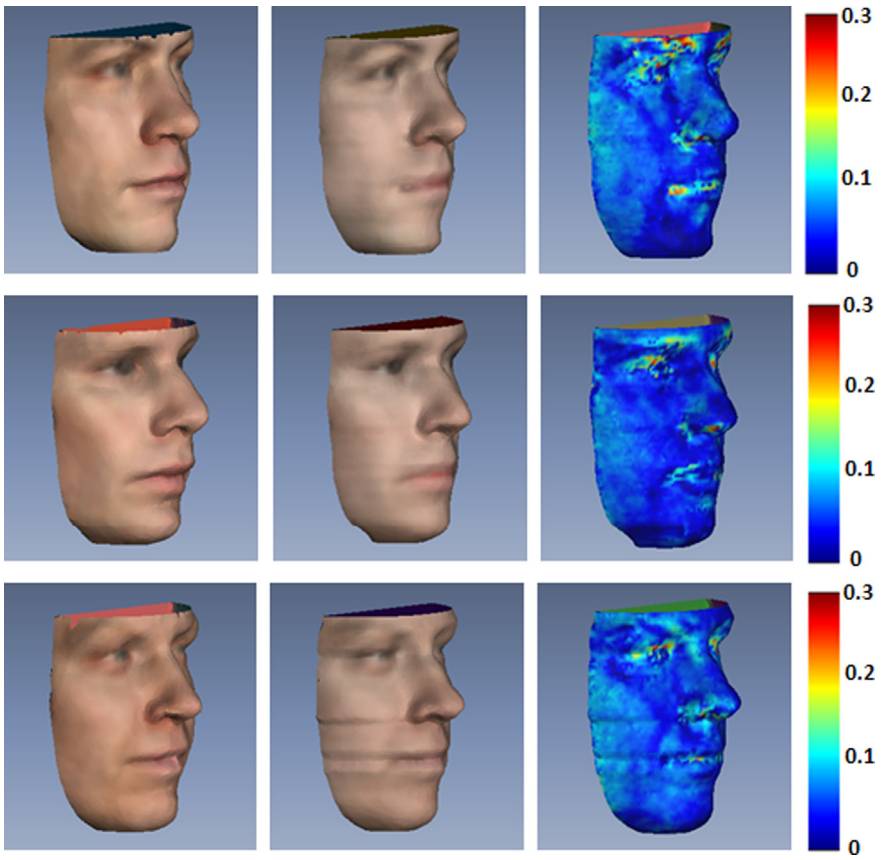


Fig. 13. Colour rendering: groundtruth (left), 3D face reconstruction $\hat{\mathbf{X}}_5$ with colour rendering (middle) and its colour error surfaces (right).

are closer to the ground truth than the mean face. Moreover, using colour information allows the reconstruction $\hat{\mathbf{X}}_5$ to be closer to the ground truth than the estimate $\hat{\mathbf{X}}_4$.

Artefacts Note that discontinuous circular bands appear on the surface of the reconstructions due to the 2D slice representation of the prior modelling (Fig. 12). However, simple post-processing like vertical smoothing can easily remove these discontinuities.

4.3.4. Rendering of the estimated colour

When using colour information, not only is the spatial location estimated but also the chrominance information at that location. Fig. 13 shows the estimated colour textures on the 3D reconstructed surface results. The texture error surface is also shown. Only chrominance information is estimated in our algorithm, so we use the intensities of the approximated prior to convert the chromaticity -red and -green into the RGB colour space for visualisation purposes only. Then the RGB colour differences between the ground truth and the reconstruction results are calculated to visualise the error surfaces. In Fig. 13, the colour textures are well estimated globally and well matched with the estimated shapes. However, it is more difficult for the methods to estimate colour information in some tiny regions of more complicated shapes and colour patterns such as parts of the mouth and the upper eye. To deal with this problem, the contour on each slice should be described with more points M to get a higher level of detail but this would increase the computation time. The Rapidform-XOR software [40] has been used for rendering the 3D reconstructions and coloured 3D reconstructions.

4.4. Discussion

We have shown first that shape from silhouettes is able to recover concavities when prior information is used for inference and second, that colour information can also be taken into account to improve the overall reconstructions. The cost functions used for inference are smooth and differentiable and are suitable for optimisation using gradient ascent techniques.

The proposed priors have been designed using standard ideas for reconstructing a contour on a basis of selected components. Depending on the nature of the database, we have proposed to compute these components either using PCA or KNN. For instance, faces (Section 4.3) are very well reconstructed with PCA with only $K = 3$ components. For comparison, similar reconstruction results have been obtained using KNN but with $K = 23$ components in this experiment.

In the first experiment (Section 4.2), KNN was the most efficient method and this can be understood by the fact that any shape in the test set will be best explained by the $K = 2$ neighbours from the prior set that correspond to the same object as viewed from a slightly different angle. For instance, the duck viewed at angle 15° in the test set is very close to the two ducks viewed at angles 0° and 30° in the training set (Fig. 6). Note that the proposed algorithms can then be adapted to any other strategy for finding the best components.

The prior is currently modelled in 2D and this can be a limitation only if the solution from the likelihood (used as an initial guess for the posterior) is not well aligned with the model in 3D space. In practise, using only silhouette information is not the best approach for processing accurate 3D reconstruction because well segmented binary silhouettes are difficult to collect and also because the optimal solution in a perfect setting is the visual hull (a convex approximation of the 3D shape and not the shape itself).

The likelihood can be improved further by taking into account more information from the sensor to recover concavities: for instance Ruttle et al. [41] used the depth information recorded by the Kinect camera to extend the KDE for the likelihood. Note that depth information also eases the segmentation of more reliable binary silhouettes for 3D reconstruction with shape from silhouettes methods.

5. Conclusion and future work

This paper has proposed KDEs of posterior density functions to infer shape from silhouettes. Optimisation is performed using gradient ascent algorithms suitable for parallel processing [18,19]. Two methods have been proposed to model the prior (PCA and KNN) and the reconstructions using these posteriors have shown that concavities can be well recovered when using prior information. The posterior has been extended to use colour information both on the likelihood and the prior. This last modelling method offers the best performance in particular when few camera views are available. Current efforts aim at extending the framework to consider other types of data (e.g. depth data) to improve the likelihood, to tackle the problem of inacurate camera parameters [41] and to investigate inference of 3D shape with prior information but without point correspondence [42].

Acknowledgments

This project has been supported by the Irish Research Council for Science, Engineering and Technology in collaboration with INTEL Ireland Ltd: funded by the National Development Plan (2008–2011) and an Innovation Partnership IP-2007-505 between Sony–Toshiba–IBM and Enterprise Ireland (2008–2010).

References

[1] C.R. Dyer, Volumetric scene reconstruction from multiple views, in: Foundation of Image Understanding, vol. 628, Springer, 2001, pp. 469–489.
[2] J.S. Franco, E. Boyer, Efficient polyhedral modeling from silhouettes, IEEE Trans. Pattern Anal. Mach. Intell. (2008) 414–427.
[3] K. Zimmermann, T. Svoboda, J. Matas, Multiview 3D tracking with an incrementally constructed 3D model, in: The Third International Symposium on 3D Data Processing, Visualization, and Transmission, 3DPVT'06, 2006, pp. 488–495.
[4] A. Laurentini, How far 3D shapes can be understood from 2D silhouettes, IEEE Trans. Pattern Anal. Mach. Intell. 17 (2) (1995) 188–195.
[5] W.N. Martin, J.K. Aggarwal, Volumetric description of objects from multiple views, IEEE Trans. Pattern Anal. Mach. Intell. 5 (2) (1987) 150–158.
[6] M. Potmesil, Generating octree models of 3D objects from their silhouettes in a sequence of images, Comput. Vis. Graph. Image Process. 40 (1) (1987) 1–29.
[7] P. Srivasan, P. Liang, S. Hackwood, Computational geometric methods in volumetric intersections for 3D reconstruction, Pattern Recogn. 23 (8) (1990) 843–857.
[8] J.-S. Franco, E. Boyer, Fusion of multi-view silhouette cues using a space occupancy grid, in: IEEE International Conference on Computer Vision, vol. 2, ICCV, 2005, pp. 1747–1753.
[9] B.G. Baumgart, Geometric modeling for computer vision, Tech. rep., Artificial Intelligence Laboratory, Stanford University, October 1974.
[10] S. Lazebnik, E. Boyer, J. Ponce, On computing exact visual hulls of solids bounded by smooth surfaces, in: IEEE Conference on Computer Vision and Pattern Recognition, vol. 1, 2001, pp. 156–161.
[11] W. Matusik, C. Buehler, L. McMillan, Polyhedral visual hulls for real-time rendering, in: Eurographics Workshop on Rendering, 2001, pp. 115–126.
[12] S. Sullivan, J. Ponce, Automatic model construction, pose estimation, and object recognition from photographs using triangular splines, IEEE Trans. Pattern Anal. Mach. Intell. 20 (1998) 1091–1096.
[13] K. Grauman, G. Shakhnarovich, T. Darrell, A Bayesian approach to image-based visual hull reconstruction, in: IEEE Computer Vision and Pattern Recognition, vol. 1, CVPR, 2003, pp. 187–194.
[14] K.N. Kutulakos, S.M. Seitz, A theory of shape by space carving, Int. J. Comput. Vis. 38 (3) (2000) 199–218, http://dx.doi.org/10.1023/A:1008191222954.
[15] A. Broadhurst, T. Drummond, R. Cipolla, A probabilistic framework for space carving, in: IEEE International Conference on Computer Vision, vol. 1, ICCV, 2001, pp. 388–393, http://dx.doi.org/10.1109/ICCV.2001.937544.
[16] D. Kim, J. Ruttle, R. Dahyot, 3d shape estimation from silhouettes using mean-shift, in: IEEE International Conference on Acoustics Speech and Signal Processing, ICASSP, 2010, pp. 1430–1433, http://dx.doi.org/10.1109/ICASSP.2010.5495474.
[17] J. Ruttle, M. Manzke, R. Dahyot, Smooth kernel density estimate for multiple view reconstruction, in: Proceedings of The 7th European Conference for Visual Media Production, CVMP, 2010, pp. 74–81, http://dx.doi.org/10.1109/CVMP.2010.17.
[18] D. Exner, E. Bruns, D. Kurz, A. Grundhöfer, O. Bimber, Fast and robust camshift tracking, in: IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, 2010, pp. 9–16.
[19] B.V. Srinivasan, Q. Hu, R. Duraiswami, Graphical processors for speeding up kernel machines, in: Workshop on High Performance Analytics-Algorithms, Implementations, and Applications, Siam Conference on Data Mining, 2010, pp. 1–9.
[20] D. Kim, 3D object reconstruction using multiple views, PhD thesis, School of Computer Science and Statistics, Trinity College Dublin, Dublin, Ireland, 2011.
[21] D. Kim, R. Dahyot, Bayesian shape from silhouettes, in: Proceedings of International Workshop on Computational Intelligence for Multimedia Understanding, Pisa, Italy, in: Lect. Notes Comput. Sci., vol. 7252, Springer Verlag, 2011, pp. 78–89.
[22] C.P. Robert, G. Casella, Monte Carlo Statistical Methods, Springer Verlag, 1999.
[23] R. Dahyot, J. Ruttle, Generalised relaxed Radon transform (GR2T) for robust inference, Pattern Recogn. 46 (3) (2013) 199–218, http://dx.doi.org/10.1016/j.patcog.2012.09.026.
[24] C. Pintavirooj, M. Sangworasil, 3D shape reconstruction based on Radon transform with application in volume measurement, in: 10th International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision, 2002, pp. 1–4.
[25] S.M. Seitz, B. Curless, J. Diebel, D. Scharstein, R. Szeliski, A comparison and evaluation of multi-view stereo reconstruction algorithms, in: IEEE Conference on Computer Vision and Pattern Recognition, vol. 1, 2006, pp. 519–528.
[26] M.S. Drew, G.D. Finlayson, S.D. Hordley, Recovery of chromaticity image free from shadows via illumination invariance, in: IEEE International Conference on Computer Vision Workshop on Color and Photometric Methods in Computer Vision, 2003, pp. 32–39.
[27] D. Berwick, S. Lee, A chromaticity space for specularity, illumination color- and illumination pose-invariant 3D object recognition, in: IEEE International Conference on Computer Vision, 1998, pp. 165–170.
[28] J. Ruttle, Statistical framework for multi-sensor fusion and 3D reconstruction, PhD thesis, School of Computer Science and Statistics, Trinity College Dublin, Dublin, Ireland, 2012.
[29] T. Iwasa, T. Shima, M. Sai, G. Xu, 3D eigenfaces for face modeling, in: The 5th Asian Conference on Computer Vision, 2002, pp. 23–25.
[30] C. Xu, Y. Wang, T. Tan, L. Quan, A new attempt to face recognition using 3D eigenfaces, in: Asian Conference on Computer Vision, vol. 2, 2004, pp. 884–889.
[31] I. Matthews, S. Baker, Active appearance models revisited, Int. J. Comput. Vis. 60 (2) (2004) 135–164.
[32] T.F. Cootes, G.J. Edwards, C. Taylor, Active appearance models, IEEE Trans. Pattern Anal. Mach. Intell. (2001) 681–685.
[33] T.F. Cootes, C.J. Taylor, D.H. Cooper, J. Graham, Active shape models – their training and application, Comput. Vis. Image Underst. 61 (1) (1995) 38–59.
[34] G.J. Edwards, A. Lanitis, C.J. Taylor, T.F. Cootes, Statistical models of face images – improving specificity, in: British Machine Vision Conference, 1996, pp. 765–774.
[35] M. Turk, A. Pentland, Face recognition using eigenfaces, in: IEEE Conference on Computer Vision and Pattern Recognition, 1991, pp. 586–591.
[36] S. Lefebvre, H. Hoppe, Parallel controllable texture synthesis, ACM Trans. Graphics (TOG) 24 (3) (2005) 777–786.
[37] J.M. Geusebroek, G.J. Burghouts, A.W.M. Smeulders, The Amsterdam library of object images, Int. J. Comput. Vis. 61 (1) (2005) 103–112.
[38] B. Moghaddam, A. Pentland, Probabilistic visual learning for object recognition, IEEE Trans. Pattern Anal. Mach. Intell. 19 (7) (1997) 696–710.
[39] P. Paysan, R. Knothe, B. Amberg, S. Romdhani, T. Vetter, A 3D face model for pose and illumination invariant face recognition, in: 6th IEEE International Conference on Advanced Video and Signal based Surveillance, AVSS, 2009, pp. 296–301.
[40] Rapidform-XOR, http://www.rapidform.com, 2010.
[41] J. Ruttle, C. Arellano, R. Dahyot, Extrinsic camera parameters estimation for shape-from-depths, in: 20th European Signal Processing Conference, EUSIPCO, Bucharest, Romania, 2012, pp. 1985–1989.
[42] C. Arellano, R. Dahyot, Shape model fitting algorithm without point correspondence, in: 20th European Signal Processing Conference, EUSIPCO, Bucharest, Romania, 2012, pp. 934–938.

Donghoon Kim received a Master degree in Computer Science in 2003 from Sungkyunkwan university, Republic of Korea. Donghoon joined Trinity College Dublin in Ireland as a PhD candidate, in 2008, and he was awarded a PhD in 2011. He is now working with Samsung electronics in

Republic of Korea. His research interests include object detection, reconstruction and recognition.

Jonathan Ruttle received a Master degree in Computer Science in Interactive Entertainment Technologies, in 2008, from Trinity College Dublin, Ireland. Jonathan gained a PhD from Trinity College Dublin in 2012. He is now working with Hewlett Packard, in Ireland. His research interests include multiple view 3D reconstruction and statistical inference from depth cameras.

Rozenn Dahyot received Master degrees in Physics (1998, Telecom Physique Strasbourg) and Image Processing (1998, University Louis Pasteur Strasbourg), in France. She gained her PhD, in 2001, from the University Louis Pasteur Strasbourg. From 2002 to 2005, she was a research fellow in Trinity College Dublin, Ireland, and University of Cambridge, UK. Since 2005, she is an Assistant Professor in the School of Computer Science and Statistics, in Trinity College Dublin. Her research interest includes robust statistics, pattern detection and recognition, image and video analysis.