International Journal of Computer Vision

Regressing Local to Global Shape Properties for Online Segmentation and Tracking

--Manuscript Draft--

Manuscript Number:	
Full Title:	Regressing Local to Global Shape Properties for Online Segmentation and Tracking
Article Type:	SI: BMVC 2011
Keywords:	Occlusion recovery; Incremental learning; Level-set based tracking; Discrete Cosine Transform
Corresponding Author:	Carl Yuheng Ren Oxford, UNITED KINGDOM
Corresponding Author Secondary Information:	
Corresponding Author's Institution:	
Corresponding Author's Secondary Institution:	
First Author:	Carl Yuheng Ren
First Author Secondary Information:	
Order of Authors:	Carl Yuheng Ren
	Victor Prisacariu
	Ian Reid
Order of Authors Secondary Information:	
Abstract:	We propose a novel regression based framework that uses online learned shape information to reconstruct occluded object contours. Our key insight is to regress the global, coarse, properties of shape from its local properties, i.e. its details. We do this by representing shapes using their 2D discrete cosine transforms and by regressing low frequency from high frequency harmonics. We learn this regression model using Locally Weighted Projection Regression (LWPR) which expedites online, incremental learning. After sufficient observation of a set of unoccluded shapes, the learned model can detect occlusion and recover the full shapes from the occluded ones. We demonstrate the ideas using a level-set based tracking system that provides shape and pose, however, the framework could be embedded in any segmentation-based tracking system. Our experiments demonstrate the efficacy of the method on a variety of objects using both real data and artificial data.

Noname manuscript No.
(will be inserted by the editor)

Regressing Local to Global Shape Properties for Online Segmentation and Tracking

Carl Yuheng Ren · **Victor Prisacariu** · **Ian Reid**

Received: date / Accepted: date

Abstract We propose a novel regression based framework that uses online learned shape information to reconstruct occluded object contours. Our key insight is to regress the global, coarse, properties of shape from its local properties, i.e. its details. We do this by representing shapes using their 2D discrete cosine transforms and by regressing low frequency from high frequency harmonics. We learn this regression model using Locally Weighted Projection Regression (LWPR) which expedites online, incremental learning. After sufficient observation of a set of unoccluded shapes, the learned model can detect occlusion and recover the full shapes from the occluded ones. We demonstrate the ideas using a level-set based tracking system that provides shape and pose, however, the framework could be embedded in any segmentation-based tracking system. Our experiments demonstrate the efficacy of the method on a variety of objects using both real data and artificial data.

1 Introduction

In recent years, there has been substantial research in segmentation based tracking, in such works as Yilmaz et al (2004); Bibby and Reid (2008); Mirmehdi et al (2009), etc. These methods extract an active contour at each frame and use it to update the shape of a tracked object. Such methods result in the efficient tracking of previously unseen objects. However, a challenge that remains is occlusion, because, unless the shape being tracked is constrained in some way, the resulting contour will have an incorrect shape. Our aim in this paper, then, is to show how to learn the set of legal shapes of a potentially deformable object *incrementally* and *online* and then how to use this learned model to detect occlusion and recover the original shape of the object at each frame. As contours form holes, break up and merge, many works favour the implicit representation of shape over the explicit one (Cremers et al, 2007). This is most often done by embedding the 2D explicit contour as the zero level of a level set embedding function (Osher and Sethian, 1988). We focus on level set based segmentation but the concepts could be applied to any other types of segmentation.

A typical solution to recover the complete shape in the presence of occlusions is to put constraints on the minimization of the level set energy function. Such methods roughly fall into two categories: the first category comprises of methods which try to capture the variance in the space of legal embedding functions (e.g. Leventon et al (2000); Tsai et al (2003); Rousson and Paragios (2002); Cremers et al (2004); Dambreville et al (2008); Prisacariu and Reid (2011b)). This was first attempted by Leventon et al (2000), where PCA was used to learn the space of zero level set embedding functions. To segment the image, the contour was evolved by minimising an energy function which combined three terms, one for the image data, one for the shape and one for the pose. The minimisation sought alignment of the curve with the image gradients, while at the same time maximising the probability of the shape. Replacing edges with regions as the main source of image information, the method of Leventon et al (2000) was extended by Tsai et al (2003). Here PCA is again used on level set functions, but the energy function can be, for example, the region based one of Vese and Chan (2002). The minimisation is done directly in the shape space i.e. by differentiation with respect to the position in

Carl Yuheng Ren
Department of Engineering Science, University of Oxford, UK
E-mail: carl@robos.ox.ac.uk

Victor Prisacariu
Department of Engineering Science, University of Oxford, UK
E-mail: victor@robos.ox.ac.uk

Ian Reid
Department of Engineering Science, University of Oxford, UK
E-mail: ian@robots.ox.ac.uk

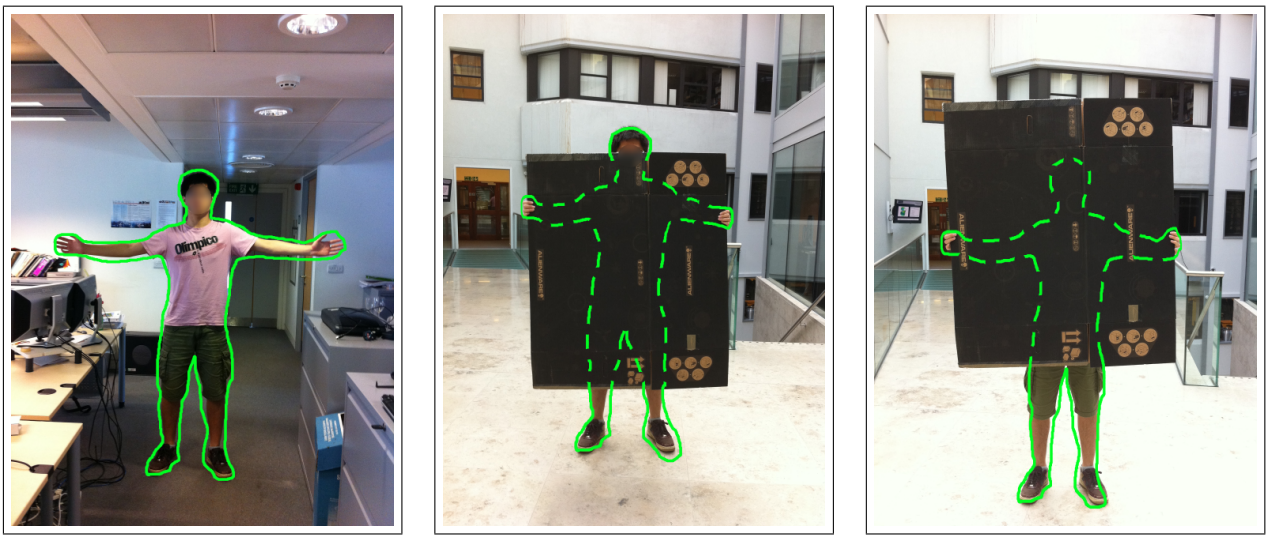

Fig. 1 Left: full human shape, from which we learn the relationship between local properties and global ones. Middle & Right: when occlusion happens, we can reconstruct the global shape from observed local properties based on the learnt relationship. (This an illustrative example of our idea, for real examples, please refer to Figure 7 and 6)

the lower dimensional latent space and the 2D pose. Non-linear dimensionality reduction was used first by Rathi et al (2006) and Dambreville et al (2008), in the form of Kernel PCA (Schölkopf et al, 1998). This has been shown to greatly improve the learning capabilities of the shape space. Here a segmentation quality measure is maximised in the space of embedding functions, aiming to minimise the distance between the projection of those embedding functions and the known lower dimensional latent points. The opposite is true for Prisacariu and Reid (2011b,a), where the optimisation process is kept in the lower dimensional space and a closed form generative process is used to generate high dimensional shapes. This is achieved by replacing Kernel PCA with Gaussian Process Latent Variable Models (Lawrence, 2005). Prisacariu and Reid (2011a) represent shapes explicitly using elliptic Fourier descriptors (Kuhl and Giardina, 1982) and generate the level set embedding functions at runtime, whereas Prisacariu and Reid (2011b) learn spaces of level set embedding functions directly, compressed using the discrete cosine transform (DCT, Watson (1994)). All of the above presented methods are robust to occlusions, since the evolution of the contour is limited to the space of possible shapes. None of them however explicitly considers occlusion modelling or recovery. Furthermore, inference tends to be very slow (in the order of seconds or minutes per frame) and training is always done *off-line*. This means that, when new contours are added, the *whole model* must be re-trained, an operation that can take up to several hours.

The second school of methods attempts to influence the shape in the current frame by comparing it with a number of recently observed shapes. Mirmehdi et al (2009) proposed to incrementally builds a dynamic space of good shape hypotheses from all frames leading up to the current one. The shape of the current frame is thus constrained by minimizing its distance from a locally Gaussian weighted shape expectation of the learned space. By continually updating a weight matrix, this method can incrementally update the space of good shapes without re-training. However, in practice, (i) both the size of the weight matrix and the time it takes to update it grows as n^2 (where n is the number of observed good shapes), and (ii) in order to keep track of this matrix, all previously observed shapes need to be stored. Alternatively, when using a fixed size weight matrix, the method suffers from rapid forgetting. The authors also note that this method is very slow, making it unsuitable for real-time operation. In another work, Yilmaz et al (2004), a dense level set function is embedded in the shape, with the background area set to zero. A probability distribution for each grid point on the level set is modelled with a single Gaussian, which is updated only where no occlusion is present. Once occlusion is detected (using area and distance heuristics), the method uses the Gaussian model on each grid point to cast an expansion force on the level set, to recover the missing parts. However, the updating rate is difficult to tune when the shape of a deformable object is learned: updating too quickly will result in recovering the current shape simply based on the previous few shape, while updating too slowly will suffer from large uncertainty.

In this paper, we consider the problem of occlusion detection and shape recovery using a different approach, by modelling the relationship between the local and global properties of shape. The motivation behind our idea is illustrated in Figure 1, where we show an occluded human (with only the legs visible). Even though the bulk of the person is occluded, a human observer can reconstruct the shape (i.e. the global property) from the relationship between the hands,

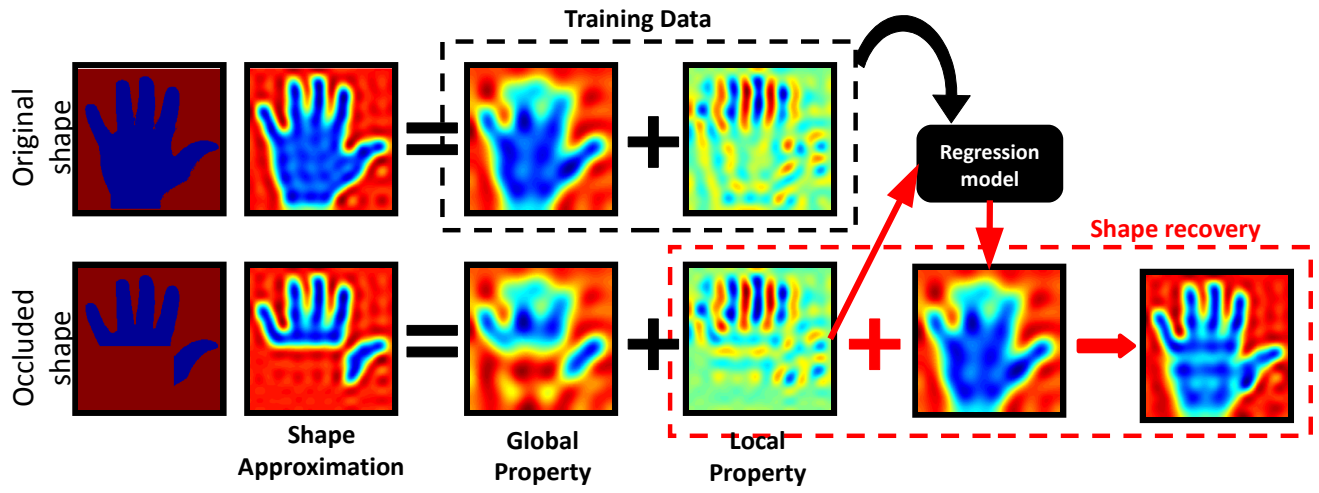

Fig. 2 An overview of our whole shape recovery algorithm.

arms, legs etc. (i.e. the local properties). We describe a method to formalize this insight by learning the relationship between the local and global properties of shapes. Specially, we show how Locally Weighted Projection Regression (LWPR, Vijayakumar et al (2005)) can be used to learn a regression from the high frequency harmonics to the low frequency ones of a shape, and how this regression can be used to detect and recover occlusions on previously seen shapes. We link our shape regression to the pixel-wise posteriors (PWP) level set-based tracker of Bibby and Reid (2008). The PWP tracker obtains the target pose (a 6 DoF 2D affinity or 4 DoF 2D similarity transform) and figure/ground segmentation at each frame. We use the pose obtained from the tracker to align the shapes and then add them to the learning framework, as they are received. After a burn-in period, the framework is able to recover occluded shapes at real time.

An early version of this work was presented as the conference paper Ren et al (2011). In the present paper we elaborate this work, provide further algorithm details, extend the literature review and add further qualitative experiments.

The remainder of the paper is structured as follows: we begin in Section 2 by discussing the discrete cosine transform shape representation and its advantages. In Section 3, Section 4 and Section 5 we detail the LWPR-DCT algorithm and describe how we detect occlusion, discriminate between occlusion and a new shape, and recover occluded shapes. We show qualitative and quantitative evaluations of our method in Section 6, and conclude in Section 7.

2 Shape representation via DCT

The 2D discrete cosine transform (DCT, Watson (1994)) is a special case of the discrete Fourier transform, which represents an image using a series of orthogonal cosine basis functions known as harmonics, each with its own frequency and amplitude. A common use for the DCT is image compression, it being the basis for the JPEG format. Similarly, in Prisacariu and Reid (2011c), the authors used it to compress level set embedding functions. Our work is based on a different property of the DCT, namely the fact that the low frequency harmonics contain the coarse "bulk" properties of the information in the signal, while high frequency ones contain the "details". When applied to shapes, this means that, often, when an object is occluded, parts of its main body may be missing, but many high frequency details remain. Our experiments suggest that occlusions introduce relatively minor changes to the high frequency DCT coefficients. Based on this observation, we train a regression model from higher frequency harmonics to lower frequency harmonics using previously observed complete shapes. Thus, when an occluded shape is observed, we extract its high-frequency harmonics and use the regressor to determine the expected low frequency harmonics, and hence recover the whole shape by adding the low frequency harmonics to the high frequency ones. Figure 2 gives an overview of our framework.

The 2D DCT of a $N \times N$ image is defined as:

$$C(u,v) = \alpha(u)\alpha(v) \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x,y) \cos\left[\frac{\pi(2x+1)u}{2N}\right] \cos\left[\frac{\pi(2y+1)v}{2N}\right]$$

for $u,v \in 0,1,2,\ldots,N-1$, $\alpha(u)$ and $\alpha(v)$ are defined as:

$$\alpha(u) = \begin{cases} \sqrt{\frac{1}{N}} \text{ for } u = 0 \\ \sqrt{\frac{2}{N}} \text{ for } u \neq 0 \end{cases}$$

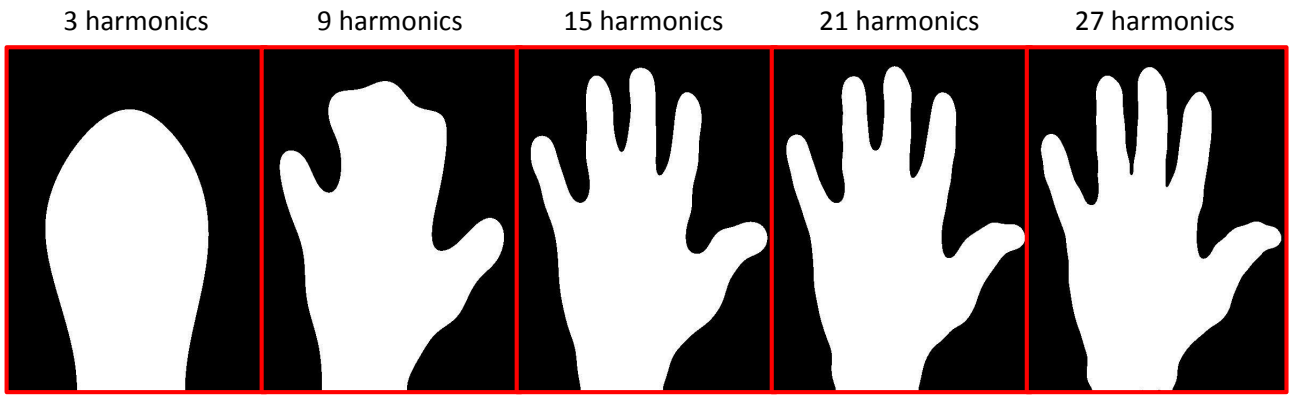

Fig. 3 Using different number of harmonics to approximate shape: as the number of harmonics that are used to encode the shape increases, more details of the shape are captured. Only the first few tens of harmonics are sufficient to recover full information of the shape, higher frequency harmonics contains more image noise than information of the shape.

The inverse transform is defined as:

$$f(x,y) = \sum_{x=0}^{N-1}\sum_{y=0}^{N-1} \alpha(u)\alpha(v)C(u,v)\cos\left[\frac{\pi(2x+1)u}{2N}\right]\cos\left[\frac{\pi(2y+1)v}{2N}\right]$$

for $x,y \in [0,\ldots,N-1]$. Since the basis functions are orthogonal, the coefficients can be computed independently, as above. The transform yields a natural hierarchical representation of the original image in which the top-left, low frequency coefficients in the DCT capture the overall signal, while the high frequency coefficients (further away from top-left) capture the details of the image .

We use the DCT to represent a silhouette mask image, i.e. a binary image of the figure/ground segmentation, with 1 for foreground and -1 for background. This is in contrast with works such as Prisacariu and Reid (2011b) where the DCT is computed from the level set embedding function. Our representation insures that the high frequency harmonics capture variability *in the contour* rather then, for example, in the structure of the level set embedding the contour.

Note that, the first several tens of harmonics are sufficient to recover the whole shape, very high frequency harmonics contains far more noise than information of the shape. For example, as is shown Figure 3, when we increase the number of harmonics from 3 to 15, we observe significant increase in the shape detail, however, from 15 to 27, very little detail information are added to the recovered shape. Thus, we use only the 10-15 DCT harmonics (depending on complexity of the shape) to encode the shape, and divide the selected harmonics into low-frequency ones and high frequency ones. Taking the inverse DCT transform of the selected harmonics then thresholding at zero yields an approximation for the silhouette.

3 Locally Weighted Projection Regression(LWPR)

In this work we aim to recover the missing part of a shape in a discriminative manner, by using an online trained regression (as opposed to learning a generative shape space) from the high frequency DCT coefficients to the low frequency ones. We therefore need to learn an incremental approximation of a highly nonlinear and high dimensional function. Established method for fitting non-linear functions globally already exists, a few examples being Support Vector Machine Regression (SVMR, Smola and Schölkopf (2004)), Gaussian Process Regression (GPR, Rasmussen (2006)) and Variational Bayes Mixture Models (VBM, Corduneanu and Bishop (2001)). These methods are however not generally suited for online learning in high dimensional spaces. First, they require a priori determination of the right function space, in terms of basis or kernel functions (GPR, SVMR) or number of latent variables (VBM). Second, all these methods are developed primarily for off-line batch training, rather than for incremental learning. For instance, in the case of SVMR, when a new training point is added, the outcome of the global optimisation can be changed greatly.

We use Locally Weighted Projection Regression (LWPR, Vijayakumar et al (2005)) as our regression model. LWPR is a nonlinear function approximator that learns rapidly from incrementally acquired data, without needing to store the training data. The computational complexity grows linearly with the number of inputs. LWPR can also deal with a large number of possibly redundant inputs, which is often the case when tracking rigid objects.

Each processing unit in LWPR is a multi-input, single-output regression model. We therefore use multiple processing units to formulate our multi-input, multi-output shape regression (LWPR-DCT model). LWPR is based on the hypothesis that high dimensional data are characterised by locally low-dimensional distribution. A learned LWPR unit has K local models, each comprising a Receptive Field (RF)

Table 1 Legend of indexes and symbols used

Notation	Affectation
M	No. of training data points
N	Input dimensionality (dim of $\mathbf{x}^{hf}$)
H	Output dimensionality (dim of $\mathbf{x}^{lf}$)
$k = 1:K$	No. of local models
$r = 1:R$	No. local projections in each local model
$\{\mathbf{x}_i^{hf}, \mathbf{x}_i^{lf}\}_{i=1}^{M}$	Training data
$\{x_{i,j}^{lf}\}_{j=1}^{H}$	Element of $\mathbf{x}_i^{lf}$
$\{\mathbf{s}_i\}_{i=1}^{M}$	Lower dim projection of input data $\mathbf{x}_i^{hf}$
$\{s_{i,r}\}_{r=1}^{R}$	Element of $\mathbf{s}_i$
$\mathbf{e}$	Prediction error in LWPR
$\{e_j\}_{j=1}^{H}$	Elements of $\mathbf{e}$
$\mathbf{c}_k$	Field centre of the k-th RF
$\mathbf{D}_k$	Distance metric of the k-th RF
$\{\mathbf{u}_r^n\}_r = 1^R$	The r-th projection of a local model after observing n training point (local model index omitted)
$\{\beta_r^n\}_r = 1^R$	The weight for the r-th projection of a local model model after observing n training point (local model index omitted)
$(\mathbf{a}_0^n, \beta_0^n)$	The means of input and output of a local regression model after n training points.
w_k	Activation of the k-th RF as in 1
W_k^n	Cumulative weights of the k-th RF after n training points
$\{SS_r^n, SR_r^n, SZ_r^n\}_{r=1}^{R}$	Trace variables for the incremental computation of the r-th local regression after seeing n training points

characterised by: (1) a field centre $\mathbf{c}_k$; (2) a positive semi-definite distance metric $\mathbf{D}_k$ that determines the size and shape of the neighbourhood contributing to the local model and (3) a local projection regression model (a modified version of partial least squares) characterised by a set of projections and respective their weights. In the following sections, we detail how we learn a LWPR-DCT model from a set of training shapes and use the learnt model to detect and recover occluded shapes.

4 Incremental online learning with LWPR

The notations used in this and the following sections are listed in Table 1. The learning algorithm of a single LWPR unit is summarised in Table 2. Learning of a LWPR unit comprises of (1) the incremental computation of projections and regressions in each local models and (2) the adjustment of the shapes and the sizes of the receptive fields (RFs). We start with the algorithm for updating the projections and regressions in each local models. Initialized with no RF, when a training shape is observed, we compute its high frequency and low frequency DCT coefficients as input and output to LWPR. Given a set of high frequency DCT coefficients $\mathbf{x}_i^{hf}$, each local model in a LWPR unit computes a RF weight, which is also known as the activation:

$$w = \exp\left(-\frac{1}{2}(\mathbf{x}_i^{hf} - \mathbf{c})^T \mathbf{D}(\mathbf{x}_i^{hf} - \mathbf{c})\right) \quad (1)$$

where we omitted the local model index k for clarity.

If no RF is activated by more than w_{gen}, a new RF centered at $\mathbf{x}_i^{hf}$ will be created with the initial distance metric $\mathbf{D}_{def}$ and two projections. $w_{gen} \leq 1$ is a threshold that determines the distance between different RFs: the closer w_{gen} is set to 1, the more overlap local models will have. $\mathbf{D}_{def}$ is a diagonal Gaussian distance metric, which determines the initial shape of the RF.

Table 2 Pseudo code for the learning of LWPR-DCT model.

- Initialize the LWPR with no receptive field (RF).
- For the i-th training shape Φ_i
  - ○ compute the first $(N+H)$ DCT coefficients of Φ_i
    - ⋆ Use No. $1 \sim H$ coefficients as output $\mathbf{x}_i^{lf}$
    - ⋆ Use No. $(H+1) \sim (N+H)$ coefficients as input $\mathbf{x}_i^{hf}$.
  - ○ For the k-th out of K existing RFs:
    - ⋆ Calculate the activation using Equation 1.
    - ⋆ Update the k-th local regression model.
    - ⋆ Update the distance metric $\mathbf{D}_k$
    - ⋆ Check the decreasing rate of MSE at each projection to see if the number of projections needs to be increased.
  - ○ If no RF was activated by more than w_{gen}:
    - ⋆ Create a new RF with initial number of projections $R = 2$ $\mathbf{c}_{K+1} = \mathbf{x}_i^{hf}$ and $\mathbf{D}_{K+1} = \mathbf{D}_{def}$, $K \leftarrow K+1$.

Each local model is initialized with input mean $\mathbf{a}_0^0 = \mathbf{0}$, output mean $\beta_0^0 = 0$ and weight $W^0 = 0$. With new training data $(\mathbf{x}_i^{hf}, x_{i,j}^{lf})$, these are updated as:

$$W^{n+1} = \lambda W^n + w \quad (2)$$

$$\mathbf{a}_0^{n+1} = (\lambda W^n \mathbf{a}_0^n + w\mathbf{x}_i^{hf})/W^{n+1} \quad (3)$$

$$\beta_0^{n+1} = (\lambda W^n \beta_0^n + w x_{i,j}^{lf})/W^{n+1} \quad (4)$$

Each LWPR local regression model is an incremental locally weighted version of partial least squares, parametrised

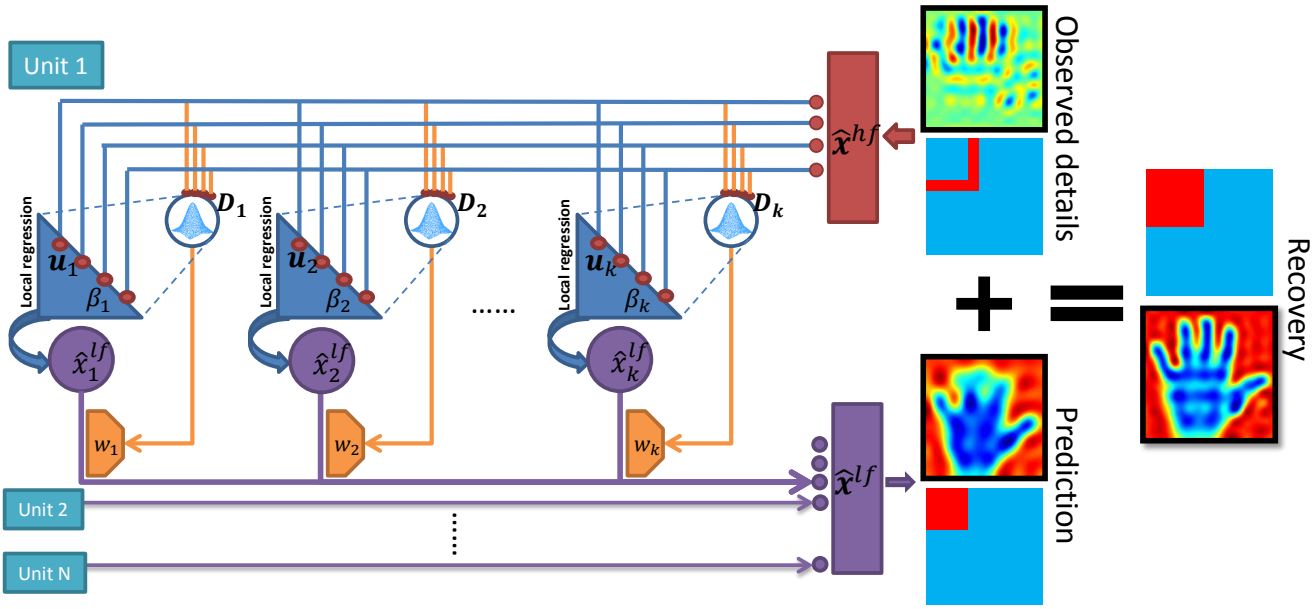

Fig. 4 Structure and work flow of LWPR, inspired by Figure 3 in Vijayakumar et al (2005).

by a set of projections $\{\mathbf{u}_r\}_{r=1}^R$ and corresponding weights $\{\beta_r\}_{r=1}^R$. Given new training data $(\mathbf{x}_i^{hf}, x_{i,j}^{lf})$, the local regression models in the j-th LWPR unit are updated as follows:

Initialize: $\mathbf{z} = \mathbf{x}_i^{hf} - \mathbf{a}_0^{n+1}$, $res_1 = x_{i,j}^{lf} - \beta_0^{n+1}$

For $r = 1 : R$

$$
\begin{aligned}
&1.\ s_{i,r} = \mathbf{z}^T \mathbf{u}_r^n / \sqrt{\mathbf{u}_r^{nT} \mathbf{u}_r^n} && (5)\\
&2.\ SS_r^{n+1} = \lambda SS_r^n + w s_{i,r}^2 \\
&3.\ SR_r^{n+1} = \lambda SR_r^n + w s_{i,r} res_r \\
&4.\ SZ_r^{n+1} = \lambda SZ_r^n + w \mathbf{z} s_{i,r} \\
&5.\ \mathbf{u}_r^{n+1} = \lambda \mathbf{u}_r^n + w \mathbf{z} res_r \\
&6.\ \beta_r^{n+1} = SR_r^{n+1} / SS_r^{n+1} \\
&7.\ \mathbf{p}_r^{n+1} = SZ_i^{n+1} / SS_r^{n+1} && (6)\\
&8.\ \mathbf{z} = \mathbf{z} - s_{i,r} \mathbf{p}_r^{n+1} \\
&9.\ res_{r+1} = res_r - s_{i,r} \beta_r^{n+1} \\
&10.\ MSE_i^{n+1} = \lambda MSE_i^n + w res_{r+1}^2 && (7)
\end{aligned}
$$

$$e_j = res_{R+1} \qquad (8)$$

where the variables SS, SR, SZ are sufficient statistics that enable use to perform incremental regression learning without the need to explicitly store any training data. $\lambda \in [0,1]$ is a forgetting factor that allows exponential forgetting of old data in the sufficient statistics.

The learning algorithm has a simple mechanism to determine whether the number of projections in a local model needs to be increased by recursively keeping track of the mean-square error (MSE, as recursively computed in Equation 7) as a function of the number of projections. If the MSE at next added projection does not decrease more than certain percentage of the previous MSE (i.e. $MSE_{i+1}/MSE_i > \phi$) the algorithm stops adding new projections.

The distance metric $\mathbf{D}$, which describes the locality of the receptive fields, can be learnt for each local model individually by stochastic gradient descent in a penalized leave-one-out cross validation cost function (indicated by the subscript $-i$):

$$J = \frac{1}{\sum_{i=1}^M w_i} \sum_{i=1}^M w_i (x_i^{lf} - \hat{x}_{i,-i}^{lf})^2 + \frac{\gamma}{N} \sum_{i,j=1}^N D_{ij}^2 \qquad (9)$$

where M denotes the number of training shapes, N is the number of RFs. γ is a trade-off parameter that can be determined empirically and the output dimension index j has been omitted for clarity.

Minimisation of Equation 9 can be accomplished in a incremental way (without the need to store any training data) as well. Vijayakumar et al (2005) proposed to expend Equation 9 with the PRESS residual error and formulated J in terms of the projected inputs $\mathbf{s}_i = [s_{i,1} \ldots s_{i,R}]^T$ in Equation 5:

$$J = \frac{1}{\sum_{i=1}^M w_i} \sum_{i=1}^M \frac{w_i (x_i^{lf} - \hat{x}_i^{lf})^2}{(1 - w_i \mathbf{s}_i^T \mathbf{P}_s \mathbf{s}_i)^2} + \frac{\gamma}{N} \sum_{i,j=1}^N D_{ij}^2 \qquad (10)$$

where $\mathbf{P}_s$ corresponds to the inverted weighted covariance matrix of the projected input $\mathbf{s}_i$ for $R = N$. Given object function Equation 10, the distance metric $\mathbf{D}$ is learnt by gradient descent:

$$\mathbf{M}^{n+1} = \mathbf{M}^n - \alpha \frac{\partial J}{\partial \mathbf{M}} \quad \text{where} \quad \mathbf{D} = \mathbf{M}^T \mathbf{M} \qquad (11)$$

where $\mathbf{M}$ is a upper triangular matrix from a Cholesky decomposition of $\mathbf{D}$. For further details on the proof and derivation of Equation 10 and Equation 11, the interested reader is referred to Appendix B in Vijayakumar et al (2005).

5 Shape Regression with LWPR-DCT

In real world applications, the proposed LWPR-DCT learning framework requires a 'burn-in' period to acquire un-occluded shapes as training data. During this period we assume the shapes adopted by an object to be un-occluded and aligned by the PWP tracker of Bibby and Reid (2008). We then transform the observed shapes into high frequency and low frequency DCT coefficients $(\mathbf{x}_{obs}^{hf}, \mathbf{x}_{obs}^{lf})$ and train a LWPR on this sequence of shapes. Once the LWPR model has been learnt, we can use it to detect occlusion in the shape as well as recover the original un-occluded shape.

Figure 4 shows the prediction workflow of a single LWPR processing unit. Given a set of novel input harmonics $\mathbf{x}^{hf}$, the prediction from a single local model follows the standard partial least squares regression:

Initialize: $\hat{x}^{lf} = \beta_0, \mathbf{z} = \mathbf{x}^{hf} - \mathbf{a}_0$

For $r = 1 : R$

1. $s = \mathbf{u}_r^T \mathbf{z}$
2. $\hat{x}^{lf} = \hat{x}^{lf} + \beta_r s$
3. $\mathbf{z} = \mathbf{z} - s\,\mathbf{p}_r$

where $\mathbf{p}_r$ is computed using Equation 6 and the local model index k omitted for clarity. The final output (i.e. a single low frequency DCT coefficient) is given by the weighted mean of all K local outputs:

$$\hat{x}^{lf} = \sum_{k=1}^{K} w_k \hat{x}_k^{lf} / \sum_{k=1}^{K} w_k \tag{12}$$

where w_k is the RF weight computed using Equation 1. Note that, since each LWPR unit is an multi-input, single-output regression model, we have the number of LWPR units equal to the number of output dimensions.

The occlusion detection and shape recovery mechanism of LWPR-DCT is summarised in Table 3: when a new shape is observed, we first compute the activation of all local models in each LWPR unit using Equation 1. If any RF in a LWPR unit has been activated by more than w_{gen} we label this LWPR unite as '*Active*'. If the number of '*Active*' LWPR units is larger than 50% of the overall number of LWPR unites, we believe the details of the shape to have been learnt before, otherwise, we classify the shape as 'not learnt' and proceed no further. For a shape whose details have been learnt, the system then makes a prediction of the low frequency components for the shape and calculate the difference between the prediction $\hat{\mathbf{x}}^{lf}$ and the observed low frequency harmonics $\mathbf{x}_{obs}^{lf}$. If the difference is smaller than twice the prediction error $\mathbf{e}$ (Equation 8) of the learnt LWPR-DCT model, we consider the shape as 'not occluded' and leave the tracker output unchanged. Otherwise, we classify the shape as being known but occluded and update it according to our prediction.

Table 3 Pseudo code for the occlusion detection and recovery using learnt LWPR-DCT model.

Given an observed shape Φ_{obs}
- ∘ compute the first H DCT coefficients of Φ_{obs}:
  - ⋆ Use No. $1 \sim (H-N)$ coefficients as $\mathbf{x}_{obs}^{lf}$.
  - ⋆ Use No. $(H-N+1) \sim H$ coefficients as $\mathbf{x}_{obs}^{hf}$.
- ∘ $count = 0$ (# of '*Active*' LWPR units)
- ∘ For the i-th out of $(H-N)$ LWPR units:
  - ⋆ Calculate the activations for all local models with Eqn 1.
  - ⋆ If any local model is activated more than w_{gen}:
    - -Label this LWPR unit as *Active*.
    - -$count \leftarrow count + 1$
- ∘ If $count \geq (H-N)/2$:
  - ⋆ Predict the low frequency harmonics $\hat{\mathbf{x}}^{lf}$ using learnt LWPR
  - ⋆ If $\|\hat{\mathbf{x}}^{lf} - \mathbf{x}_{obs}^{lf}\|^2 \geq 2\|\mathbf{e}\|^2$
    - -Shape occluded, recover shape by replacing $\mathbf{x}_{obs}^{lf}$ with $\hat{\mathbf{x}}^{lf}$.
    - else
    - -Shape is not occluded.
  - else
  - ⋆ The high frequency details have not been learnt, stop.

6 Experiments and performance analysis

We tested our method both qualitatively and quantitatively, on several video sequences and data sets. We used an Intel Core i7-870 (2.93GHz) machine to run all our experiments. We denote our method with *LWPR-DCT*.

We begin with the qualitative analysis. Examples of successful shape recovery using artificially generated occlusions are shown in Figure 5. The images on the left of each column shows the shape approximation using the inverse of the truncated DCT (red $\rightarrow$ 1, blue $\rightarrow$ -1). The images on the right show the recovered shape by thresholding the images on the left at zero level. We show the original silhouette in the left two columns, the occluded silhouette in the middle two columns and the recovered silhouette in the right two columns. The results show that LWPR-DCT is capable of recovering the shape in the presence of various types of artificially introduced occlusion.

In Figure 7 and 6, we compare our algorithm to the standard pixel-wise posteriors (PWP) tracker of Bibby and Reid (2008) on real video sequences and show that we are able to successfully recover the correct contour, in the presence of heavy occlusions. In the first 2 frames of Figure 7 there are no occlusions, so both LWPR-DCT and the standard PWP

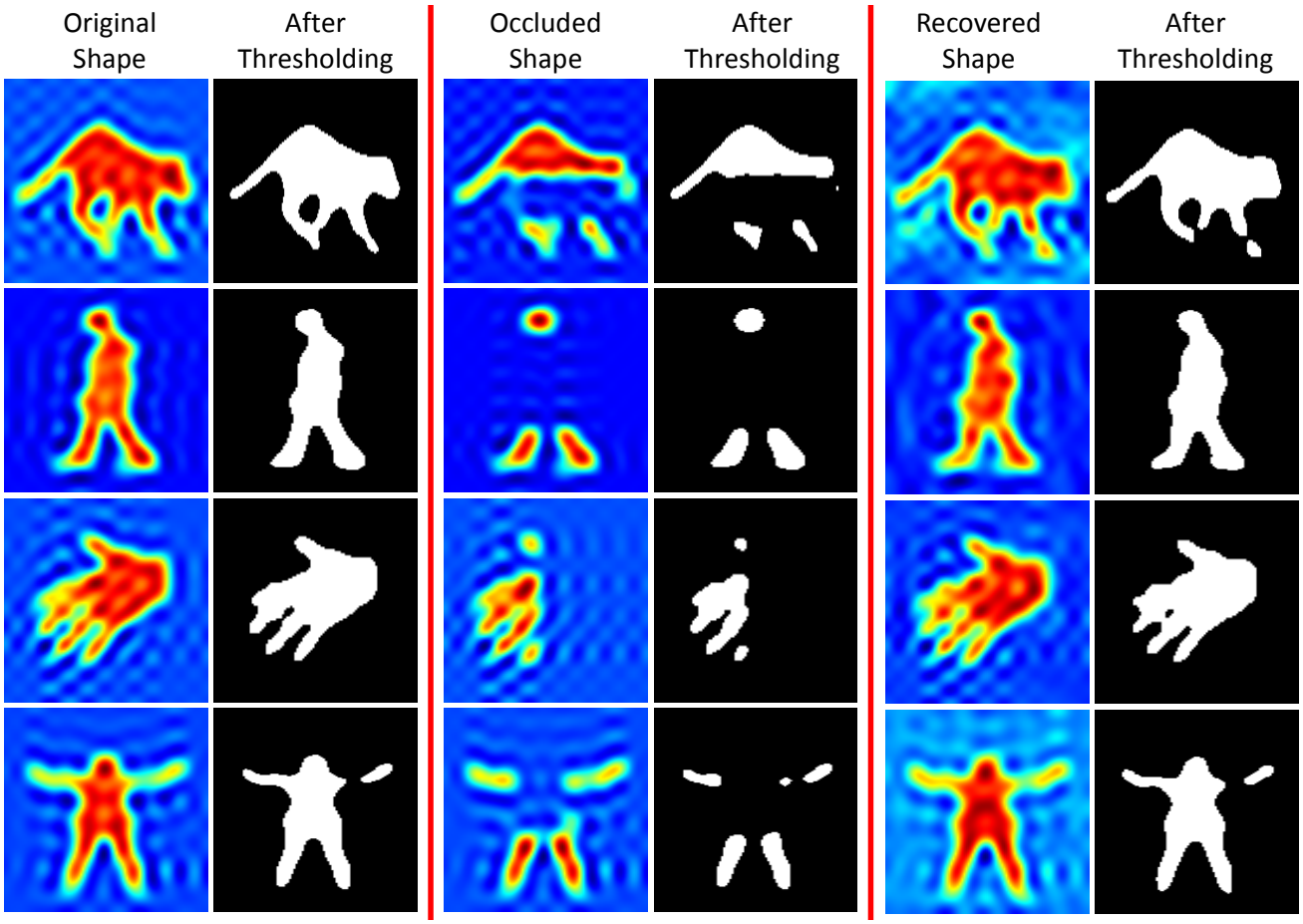

Fig. 5 Examples of recovered shapes from artificially occluded images. The left column of each pair shows false color images(blue=-1, red=1) of the inverse "truncated DCT" (i.e.. the approximation of the silhouette via the first 10 to 15 harmonics), while the right column shows the silhouette obtained from thresholding the approximation at zero. From top row to bottom row: *Cat running*, *Man walking* , *Hand*, *Woman jumping*

Fig. 6 Example frames from a video tracking a car, comparing our method to the PWP tracker of Bibby and Reid (2008). When the car is not occluded both methods produce similar results. When the tree is in front of the car the segmentation produced by the PWP tracker is corrupted, while the one produced by our tracker is not.

tracker yield similar results. When the hand is occluded, in the other 4 frames, the output segmentations of the PWP tracker are corrupted, while ours are still correct. Similarly in Figure 6, in the presence of the occlusion introduced by the tree, our LWPR-DCT framework can still recover the learnt car shape.

We show two failure cases of our method in Figure 8. The failure modes are presented using artificially introduced occlusions. LWPR-DCT can fail in two ways. First, when too many small occlusions are present, the high frequency DCT harmonics may be affected, as is shown in the upper row of Figure 8. The failure here happens because the input to LWPR-DCT has been changed considerably by the

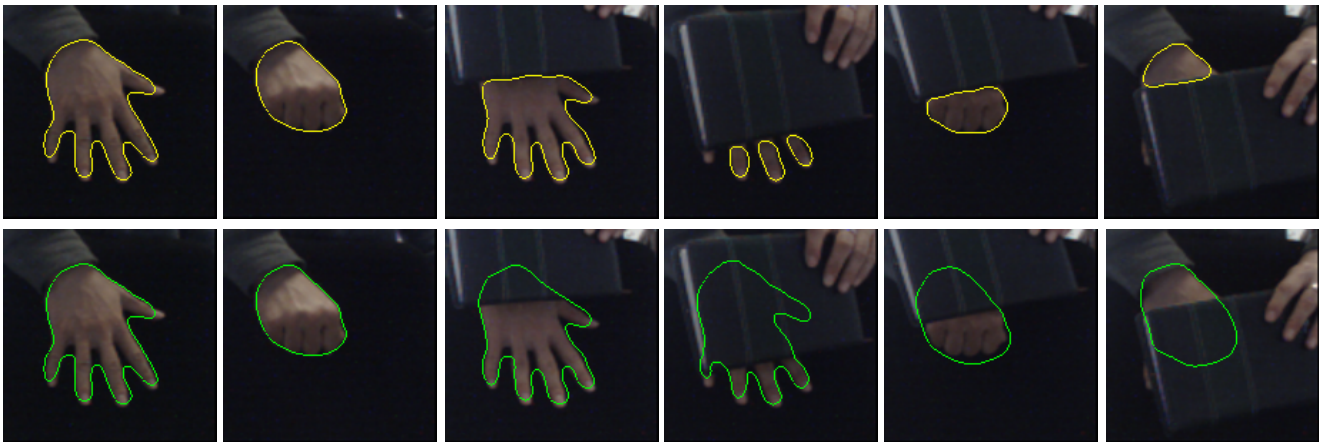

Fig. 7 Example frames from a video tracking a hand, comparing our method to the PWP tracker of Bibby and Reid (2008). When no occlusions are present, both method produce similar results. However, as soon as the hand is occluded, the PWP tracker produces an incorrect segmentation, while our method still generates correct contours.

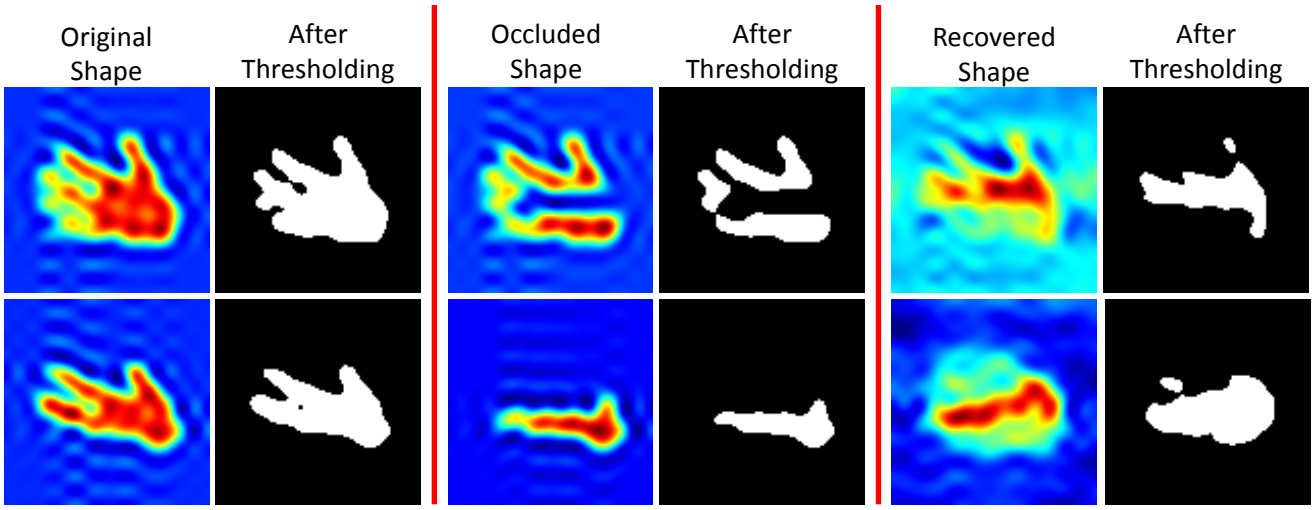

Fig. 8 Example failure cases (from the *hand* video). Top line fails because noisy high frequency harmonics are introduced, while bottom line fails because details are missing.

occlusions. In real world applications, this case is usually observed when the object being tracked is behind a fence or occluded by several small objects. The second failure case happens when too much detail is occluded, as is shown in the lower row of Figure 8. This happens because the high-frequency DCT harmonics, which we rely on, are missing. Note that, in both failure cases, the input occluded shapes may be classified as 'not learnt' in the occlusion detection stage, however, this is not guaranteed, thus LWPR-DCT might produce incorrect results.

We designed three sets of experiments to evaluate our LWPR-DCT framework quantitatively. First we measure on the effectiveness of the shape recovery using LWPR-DCT and show how many high frequency harmonics are required as input for occlusion recovery. We then evaluate the effectiveness of occlusion detection with LWPR-DCT using different number of input harmonics. Finally, we compare our algorithm with a state-of-the-art shape prior based method of Prisacariu and Reid (2011c) (denoted by GPLVM-DCT) on the performance of occlusion recovery and average processing time.

For the first two experiments, we used four datasets to evaluate the effectiveness of LWPR-DCT: *Cat running* (artificial video with few distinct poses, 398 frames), *Woman jumping* (real video with an average number of distinct poses, 410 frames), *Man walking* (real video with many distinct poses, 411 frames, the subject 2 walk of the HumanEva I dataset of Sigal et al (2010))) and *Hand* (real video with many distinct poses, 408 frames). For each video, all frames are segmented and aligned using the PWP tracker, then added to LWPR-DCT as training data. We then add different sizes of artificial occlusions (where each occlusion is rectangular and in a random location) to each frame. We chose to generate occlusions artificially in order to be able to control the

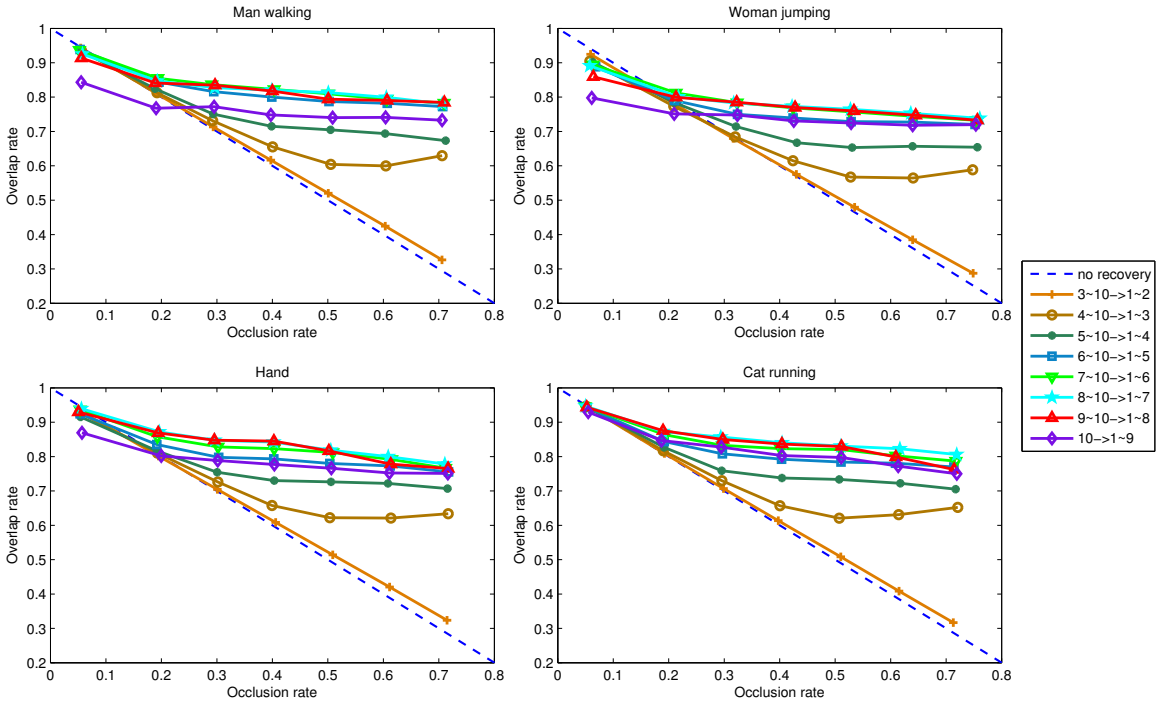

Fig. 9 Shape recovery performance evaluation of LWPR-DCT on four datasets using different number of harmonics as input and output.

percentage of occlusion and to have accurate knowledge of the ground truth. For each frame, we generate 7 levels of occlusion, ranging from 0.1 (10%) to 0.8 (80%).

In the first test, we use the overlap rate $R = \frac{S_{gt} \bigcap S_{rcv}}{S_{gt} \bigcup S_{rcv}}$ as our performance criteria, where S_{gt} is the ground truth shape and S_{rcv} is the recovered shape. We use the first 10 harmonics to approximate the segmented shapes and run tests on all possible combinations of the numbers of input and output harmonics (harmonic 10 generating 1 to 9, 9 and 10 generating 1 to 8, etc.). Figure 9 shows the results. Our method gives sensible results just by using the 10-th harmonic to regress all 1∼9 harmonics. Using the harmonics 8∼10 to regress harmonics 1∼7 or 9∼10 to 1∼8 gives the best performance in all cases. Performance decreases again as we increase the number of known harmonics. This happens because we are using too many low frequency harmonics as high frequency input, and such input as been corrupted by occlusion.

In the second test, we use the same training and test set as the first experiment, but in the testing set, we also added the original un-occluded shapes. We evaluate the effectiveness of our occlusion detection method by tracking the average precision and recall on all four data sets. We use the first 10 harmonics to approximate the shapes and run tests on all train–test combinations to see how many input/output harmonics are needed to obtain the best result. Note that, given the observed shape has been learnt before (Table 3), the occlusion detection in LWPR-DCT relies on a single threshold

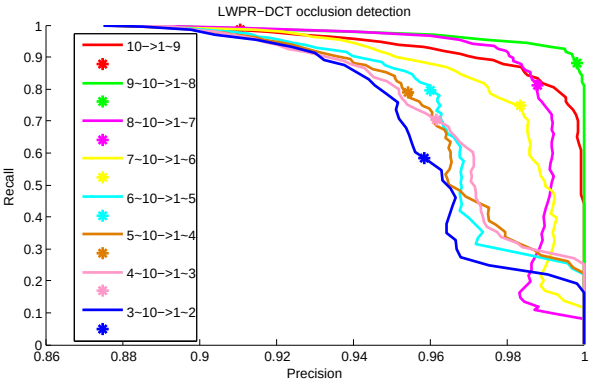

Fig. 10 Precision-recall curve of occlusion detection using LWPR-DCT, '$*$' on each curve correspond to *threshold* = **e**.

parameter, i.e. if $\|\hat{\mathbf{x}}^{lf} - \mathbf{x}^{lf}_{obs}\|^2 \geq 2\|threshold\|^2$ the observed shape will be classified as occluded. We use the prediction error **e** obtained in the training phase as this threshold (Table 3). In this experiment, only shapes that are detected as occluded are labelled as *positive*, shapes that are not learnt and shapes that are detected as un-occluded are labelled as *negative*. In Figure 10 we track the precision-recall by varying the *threshold*. Also, the point *threshold* = **e** is plotted '$*$'. As is shown in Figure 10, using the harmonics 9∼10 to regress harmonics 1∼8 gives the best performance. Using the prediction error **e** as the threshold also (approximately) gives the best precision-recall on the curve. As we increase the number of input harmonics, the detection performance de-

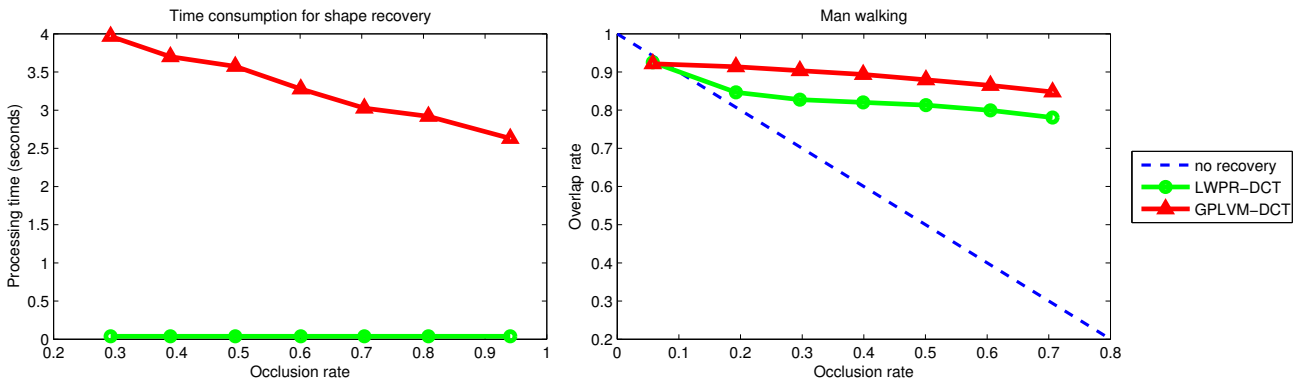

Fig. 11 Comparing LWPR-DCT to GPLVM-DCT on processing time (left) and occlusion recovery performance (right).

creases. This decrease in performance again shows that occlusions affect the low frequency harmonics more than that high frequency ones.

In the last quantitative experiment, we compare our algorithm to the shape prior method of Prisacariu and Reid (2011c), which generates embedding functions from a 2 dimensional GPLVM latent space. Here, segmentation (i.e. the recovering of the unoccluded shape), is an iterative nonlinear minimization in the learned latent space. In our experiment, for each occluded shape, we run three separate minimizations, we compute the recovery rate for each resulting shape, and we take an average of those values. We run multiple minimizations (rather than a single one) because each one can converge to a different shape, so to accurately measure the performance of Prisacariu and Reid (2011c) on our test data we need to consider all these results. As starting points for the minimization, we use the three points that generate the shapes most similar to the ground truth from the previous frame. We run both methods on the training and testing data from the *man walking* sequence from last experiment. Figure 11 shows the time consumption and recovery rate of both methods. As a well trained, shape prior based method, GPLVM-DCT outperforms our method by an average of 10%. But, as is shown in the timings chart, the time consumption for LWPR-DCT stays constant at around 35ms per shape, while the processing time required by GPLVM-DCT increases with the occlusion rate and it is much larger than LWPR-DCT (up to 114 times higher). This happens because, when using LWPR-DCT, each shape recovery is a single (closed form) regression, while, in the GPLVM-DCT case, segmentation is an iterative process with the number of iterations being proportional to the percentage of occlusion in the image. Note that the GPLVM-DCT timings shown in Figure 11 are for a single mode search. Since we use three such searches, the actual processing time per frame it three times as large. In this experiment we used the harmonics 8∼10 to regress the other 1∼7 harmonics. Note that the dot line corresponds to no-recovery, which, when less then 10% occlusion is introduced, actually has higher accuracy. This is an artefact caused by the fact that we only use a small number of DCT harmonics to represent shapes and affects the method of Prisacariu and Reid (2011b) as well. It can be easily avoided by using more DCT harmonics, at the expense of an increase in speed and memory usage.

7 Conclusions

In this paper, we have presented a novel regression based framework for online shape learning and recovery. Shapes are represented by discrete cosine transform harmonics and the set of object shapes is modelled by a regression from the high frequency harmonics to the low frequency harmonics. Our method incrementally learns a shape model for an observed object and detects/recovers occlusions in real time. We integrated our method with a level-set based tracker, but it could be potentially linked to other types of segmentation and tracking algorithms.

Our method currently has two limitations. First, the DCT representation of shape is rotation dependent, i.e small rotation change of a shape can make the high frequency coefficient change greatly, resulting in very different prediction results. Currently we rely on the PWP tracker, which obtains camera pose and segmentation at each frame, to align the shapes. Secondly, some special types of occlusion are very difficult for LWPR-DCT to handle: (1) when noisy high frequency components are introduced by small occlusion and (2) when the details of the shape are occluded. In these two cases, LWPR-DCT might give incorrect predictions, while (much slower) shape prior based methods would be more applicable.

While we have demonstrated the value of LWPR for shape recovery under occlusion, we believe that this general idea wider applications. For example, we could consider regressing local appearance to global positions, which would have similarity to Blaschko and Lampert (2008) and Fritz et al (2005), or more ambitiously regress local appearance

to global appearance. The method would also extend naturally to 3D shapes and 3D data.

References

Bibby C, Reid I (2008) Robust real-time visual tracking using pixel-wise posteriors. In: European Conference on Computer Vision, pp 831–844

Blaschko MB, Lampert CH (2008) Learning to Localize Objects with Structured Output Regression. In: European Conference on Computer Vision, pp 2–15

Corduneanu A, Bishop C (2001) Variational Bayesian Model Selection for Mixture Distributions. In: Artifical Intelligence and Statistics, pp 27–34

Cremers D, Osher S, Soatto S (2004) Kernel density estimation and intrinsic alignment for knowledge-driven segmentation: Teaching level sets to walk. In: International Journal of Computer Vision, pp 36–44

Cremers D, Rousson M, Deriche R (2007) A review of statistical approaches to level set segmentation: Integrating color, texture, motion and shape. International Journal of Computer Vision 72(2):195–215

Dambreville S, Rathi Y, Tannenbaum A (2008) A framework for image segmentation using shape models and kernel space shape priors. IEEE Transactions Pattern Analysis and Machine Intelligence 30(8):1385–1399

Fritz M, Leibe B, Caputo B, Schiele B (2005) Integrating representative and discriminant models for object category detection. In: International Conference on Computer Vision, pp 1363–1370

Kuhl F, Giardina C (1982) Elliptic fourier features of a closed contour. Computer Graphics and Image Processing 18(3):236–258

Lawrence N (2005) Probabilistic non-linear principal component analysis with gaussian process latent variable models. Journal of Machine Learning Research 6:1783–1816

Leventon M, Grimson E, Faugeras O (2000) Statistical shape influence in geodesic active contours. In: International Conference on Computer Vision and Pattern Recognition, vol 1, pp 316–323

Mirmehdi M, Chiverton J, , Xie X (2009) On-line learning of shape information for object segmentation and tracking. In: British Machine Vision Conference

Osher S, Sethian JA (1988) Fronts propagating with curvature-dependent speed: Algorithms based on hamilton-jacobi formulations. Journal of Computers 79:12–49

Prisacariu VA, Reid I (2011a) Nonlinear shape manifolds as shape priors in level set segmentation and tracking. In: International Conference on Computer Vision and Pattern Recognition, pp 2185–2192

Prisacariu VA, Reid I (2011b) Shared shape spaces. In: International Conference on Computer Vision

Prisacariu VA, Reid I (2011c) Shared shape spaces. In: International Conference on Computer Vision

Rasmussen CE (2006) Gaussian processes for machine learning

Rathi Y, Dambreville S, Tannenbaum A (2006) Statistical shape analysis using kernel PCA. In: International Society for Optical Engineering, vol 6064, pp 425–432

Ren Y, Prisacariu V, Reid I (2011) Regressing Local to Global Shape Properties for Online Segmentation and Tracking. In: British Machine Vision Conference, pp 11.1–11.10

Rousson M, Paragios N (2002) Shape priors for level set representations. In: European Conference on Computer Vision, pp 78–92

Schölkopf B, Smola A, Müller KR (1998) Nonlinear component analysis as a kernel eigenvalue problem. Neural Computation 10:1299–1319

Sigal L, Balan A, Black M (2010) HumanEva: Synchronized Video and Motion Capture Dataset and Baseline Algorithm for Evaluation of Articulated Human Motion. International Journal of Computer Vision 87:4–27

Smola AJ, Schölkopf B (2004) A tutorial on support vector regression

Tsai A, Yezzi A, Wells W, Tempany C, Tucker D, Fan A, Grimson E, Willsky A (2003) A shape-based approach to the segmentation of medical imagery using level sets. Transactions on Medical Imaging 22(2):137–154

Vese L, Chan T (2002) A multiphase level set framework for image segmentation using the mumford and shah model. International Journal of Computer Vision 50(3):271–293

Vijayakumar S, D'Souza A, Schaal S (2005) Incremental online learning in high dimensions. Neural Computation 17:2602–2634

Watson AB (1994) Image compression using the discrete cosine transform. Mathematica Journal 4:81–88

Yilmaz A, Li X, Shah M (2004) Contour-based object tracking with occlusion handling in video acquired using mobile cameras. IEEE Transactions Pattern Analysis and Machine Intelligence 26(11):1531–1536