Int J Comput Vis (2016) 118:95–112
DOI 10.1007/s11263-015-0873-2

2D-3D Pose Estimation of Heterogeneous Objects Using a Region Based Approach

Jonathan Hexner[1] · **Rami R. Hagege**[1]

Received: 26 October 2014 / Accepted: 23 November 2015 / Published online: 19 December 2015

Abstract Recently, region based methods for estimating the 3D pose of an object from a 2D image have gained increasing popularity. They do not require prior knowledge of the object's texture, making them particularity attractive when the object's texture is unknown a priori. Region based methods estimate the 3D pose of an object by finding the pose which maximizes the image segmentation in to foreground and background regions. Typically the foreground and background regions are described using global appearance models, and an energy function measuring their fit quality is optimized with respect to the pose parameters. Applying a region based approach on standard 2D-3D pose estimation databases shows its performance is strongly dependent on the scene complexity. In simple scenes, where the statistical properties of the foreground and background do not spatially vary, it performs well. However, in more complex scenes, where the statistical properties of the foreground or background vary, the performance strongly degrades. The global appearance models used to segment the image do not sufficiently capture the spatial variation. Inspired by ideas from local active contours, we propose a framework for simultaneous image segmentation and pose estimation using multiple *local* appearance models. The local appearance models are capable of capturing spatial variation in statistical properties, where global appearance models are limited. We derive an energy function, measuring the image segmentation, using multiple *local* regions and optimize it with respect to the pose parameters. Our experiments show a substantially higher probability of estimating the correct pose for heterogeneous objects, whereas for homogeneous objects there is minor improvement.

Keywords Local region-based 3D pose estimation · Three-dimensional pose estimation

1 Introduction

2D-3D pose estimation aims to determine the pose of a known 3D object from a single 2D image relative to a calibrated camera. For the case of a rigid body, its pose may be described by a 6 DOF (degrees of freedom) transformation, consisting of 3 displacement parameters and 3 rotation parameters. 3D pose estimation is commonly used as a basis for 3D tracking. 3D tracking has many applications among them—visual servoing of robotic arms, augmented reality applications such as medical visualization, entertainment and target tracking, see Lepetit and Fua (2005) for a complete survey (Fig. 1).

1.1 Motivation

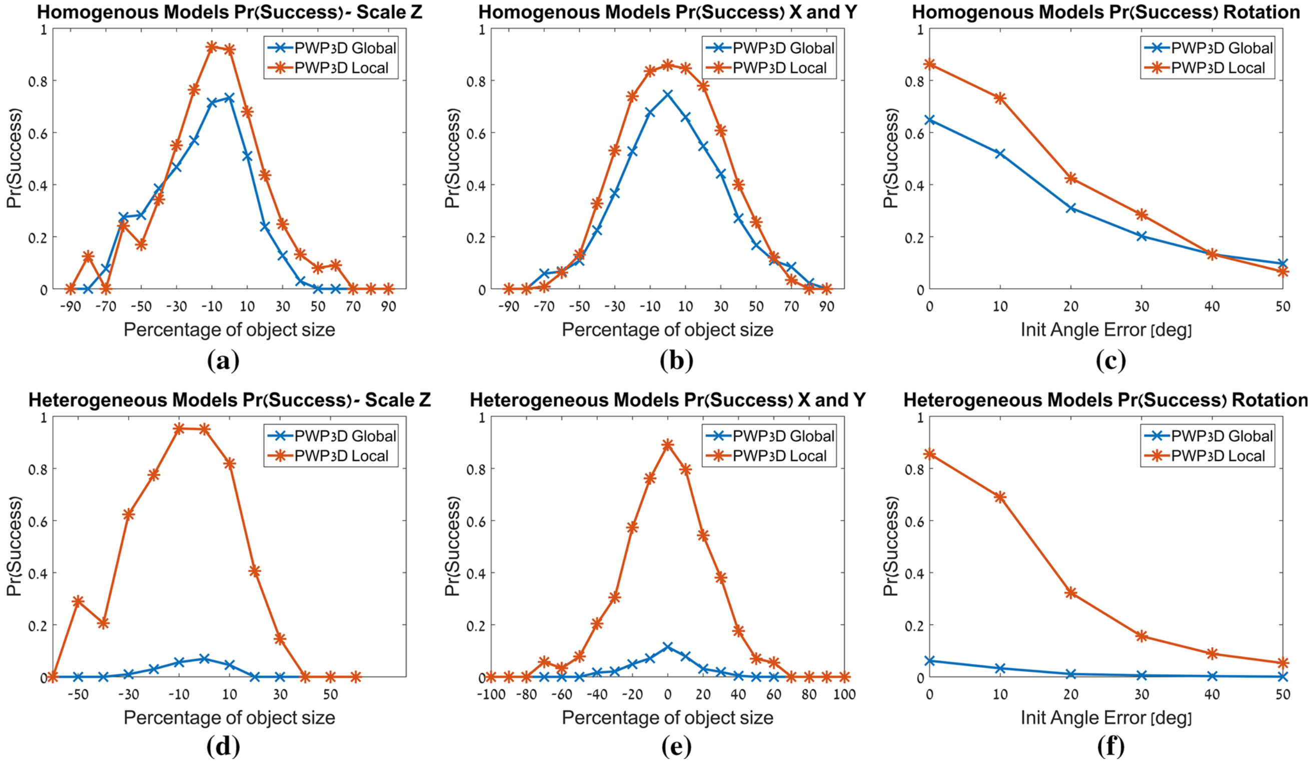
Recently, Prisacariu and Reid (2012) presented the PWP3D algorithm for simultaneous 2D-3D pose estimation and image segmentation using a known 3D model. Following the assumption that the 3D pose of the object corresponds with the optimal segmentation of the image into foreground and background, Prisacariu and Reid (2012) define an energy function which measures the quality of fit of global appearance models used to describe each one of the regions. In this context the foreground region is the projection of the object

Communicated by Radu Horaud.

✉ Jonathan Hexner
jonathan.hexner@gmail.com

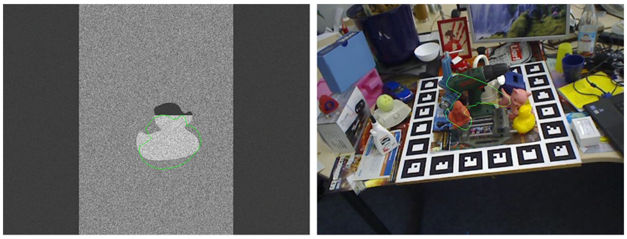
Rami R. Hagege
hagege@ee.bgu.ac.il

[1] Ben-Gurion University, Beer-Sheva, Israel



(a) **(b)** **(c)** **(d)** **(e)** **(f)**

Fig. 1 Summarized performance of analysis: probability of estimating the correct pose of homogeneous and heterogeneous objects



(a) **(b)**

Fig. 2 Pose estimation in complex scenes using global appearance models

on to the image plane, the background is the complementary region and the segmentation is a measure of the statistical fit of the pixels within each region. The appearance models, are adopted from the generative pixel-wise model presented by Bibby and Reid (2008), where the appearance models are described using posterior probability functions, rather than commonly used likelihood probability functions. Next, analytic expressions for the energy gradients with respect to the object's pose parameters are derived, and standard gradient-based minimization is applied.

The PWP3D algorithm (Prisacariu and Reid 2012) achieves state of the art performance, while keeping a low computation cost. Using a Geforce GPU with parallelizing the code, the algorithm runs at real-time. However, running the algorithm in complex scenes, containing heterogeneous objects or a cluttered background reveals a significant degradation in performances. In this context we define heterogeneity, as spatial variation in statistical properties. We demonstrate this with two examples, where the 3D pose of an object is estimated in a complex scene. In both examples the object's pose parameters are initialized to the ground truth, i.e., the algorithm begins when the object is at the correct pose, in order to avoid possible dependency on the optimization algorithm.

1. The scene in the first example (Fig. 2a) is synthetic, comprising a non-homogeneous duck and a non-homogeneous background. Most of the duck's pixels are light colored, however the top of its head has a small dark region. The background too is mostly light colored, with dark distant regions. The object's pose estimated by the algorithm is depicted by the green contour in Fig. 2a. The incorrect pose is a result of the global appearance models, used by the PWP3D algorithm, which associate the dark regions of the scene with the background, causing the duck to shift away from dark regions, placing the object in light-pixel areas.
2. In the second example (Fig. 2b) there is a heterogeneous driller object from the ACCV database (Hinterstoisser et al. 2012). The driller is mostly green, however its head is black. The background is mostly cluttered, with a significant black region at the bottom of the image. Once

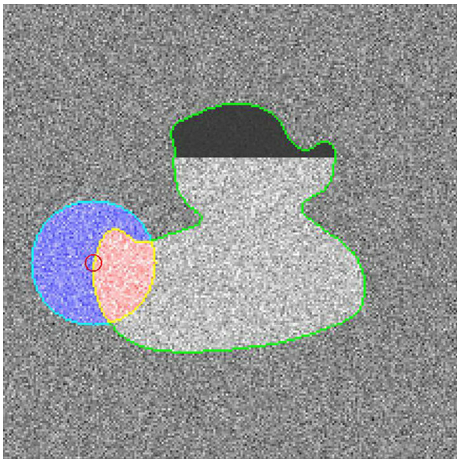


Fig. 3 Single local region

again the black pixels are associated with the background, causing the object to drift away from the correct pose. The object's estimated pose is depicted by the green contour in the image.

Theses examples show that describing complex scenes, containing significant spatial variation of statistical properties, using global appearance models does not give a sufficient description of the scene, leading to an incorrect pose estimation. We suggest applying ideas from local active contours (Lankton and Tannenbaum 2008) to develop appearance models which will capture the spatial variation in the foreground and background regions. We define multiple local regions centered around the 2D contour points. Each local region comprises a local foreground and a local background region, as illustrated in Fig. 3. In this figure a single local region, centered at one of the contour points, is shown with its separation to local foreground and local background. For each local region we define a local energy function measuring the segmentation quality within that region. Next, we define an energy function which fuses together the local energies, and which we optimize, with respect to the pose parameters. We applied the localized algorithm to the scenes in Fig. 2, where the PWP3D failed. The results using the localized appearance are shown in Fig. 4a, b. As depicted by the objects' contours the localized algorithm successfully estimates their correct pose. The local regions used are circles with a radius of 30 pixels, which are shown along the objects' contours.

We demonstrate our algorithm's improvement over the PWP3D algorithm, by measuring the basin of attraction of the rotation angle across all axes, translation in X and Y and scale across multiple homogeneous and heterogeneous models (Fig. 1). In each experiment the algorithms are applied after initializing the object's pose to a random initial error. Success is defined as a final rotational error of not more than

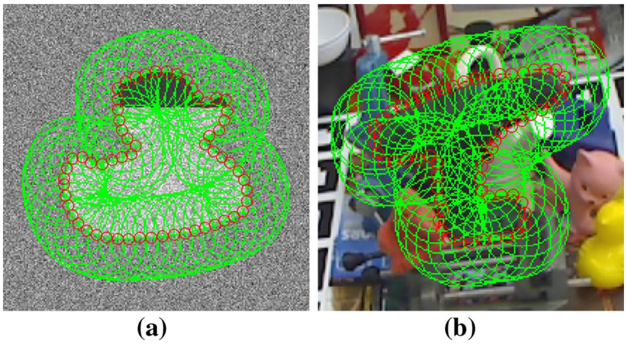


Fig. 4 Pose estimation in complex scenes using local appearance models

10 degrees and 10% of object size in translation. The probability of error, per initial error bin, is defined as the ratio between the number of cases where the pose is estimated successfully out of the total number of cases in that bin. The full details of the experiment are provided in Sect. 5. This figure shows a dramatic improvement when using the localized algorithm for heterogeneous objects, and little improvement for homogeneous objects both in translation, rotation and scale experiments.

1.2 Relation to 2D-3D Pose Estimation Approaches

The 3D pose estimation and tracking literature is very extensive, and exceeds the scope of this paper, we present a short review of the existing methods. We follow the approach of Lepetit and Fua (2005) and divide the approaches into two major types:

1. Edge based methods—these methods match the 3D object's projected edges with those in the image.
2. Methods which rely on information inside the object's projection.

The edge based methods may rely on strong gradients in the image without explicitly extracting contours e.g., the RAPiD tracker of Harris and Stennet (1990) or on explicit contours of the object e.g., Lowe (1987). The main drawback of this approach is numerous local minima (Brox et al. 2010), and sensitivity to noise or missing information (Dambreville et al. 2010).

Methods which rely on information inside the object's projection include:

- Methods which rely on local interest points—e.g., SIFT (Scale Invariant Feature Transform) points of Lowe (2004).SIFT points possess very important attributes: they are invariant to scale, rotation and constant illumination changes. The majority of 3D pose estimation and 3D object recognition work (e.g. Arie-Nachimson and Basri

2009; Savarese and Li 2007) performed today relies on these features. However, these features capture mostly the texture of the scene. This could be problematic in several scenarios: in the case where the object's texture is not known a-priori or changes in the scene (e.g. due to dirt), or in the case of a texture-less object.

- Region-based approaches, which is the focus of this paper, assume that the object's pose corresponds with the optimal segmentation into foreground and background. Using a known 3D model the foreground region is defined as the object's silhouette and the complementary region as the background. Region based methods are well proven for active contour segmentation of the foreground from the background (e.g., Chan and Vese 2001). The foreground is referred to as the interior of a contour, whereas the background is the exterior of the contour. Typically, an energy functional is defined, comprising the quality of fit of each of the regions and a penalty term, limiting the contour length. The contour is found by iteratively propagating each point in the direction which optimizes the energy functional.

More recently, with the increased availability of cameras capable of depth measurement, several methods have been proposed (Tan and Ilic 2014; Brachmann et al. 2014) using depth information.

1.3 Region Based 2D-3D Pose Estimation

Several approaches have been suggested in the context of combined image segmentation and 3D pose estimation or 3D tracking.

Rosenhahn et al. (2007) extend the classical region-based segmentation energy by a 2D shape similarity term, which measures the distance between the evolved curve and the projection of the 3D shape onto the image plane. This term restricts the contour propagation to the vicinity of the object's contour. Every iteration comprises two main stages:

1. The curve is propagated in the direction which optimizes the energy function.
2. A correspondence between the 3D model and the curve is estimated, in terms of a 6 DOF transformation. The transformation is applied to the 3D model and the curve is reinitialized according to the 3D projected curve.

In a later version, Schmaltz et al. (2007) simplify the calculations by eliminating the contour propagation stage and performing the optimization directly with respect to the pose parameters. The energy functional is computed based on classical region based terms, without the shape similarity term. Every point along the contour is assigned a force in the direction normal to the curve. The sign of the force, exterior or interior to the curve, is determined according to the region achieving a better energy value. Using 2D-3D point correspondences the 2D force is translated to a 3D force direction, which results in a rigid body transformation. Later, Schmaltz et al. (2009) suggested an integrated tracking system comprising the region based approach from Rosenhahn et al. (2007), complemented by a 2D SIFT tracker and optical flow for motion estimation between frames.

Dambreville et al. (2010) define an energy function in terms of the 3D surface model, which is assumed to be known, and its pose parameters. In contrast to Schmaltz et al. (2007), Dambreville et al. (2010) use differential geometrical tools to calculate the gradient of the energy with respect to the pose parameters, and propagate the object's pose in this direction. This approach has a strong advantage as it allows propagating the pose parameters in the direction of the optimal segmentation. Their algorithm consists of the following steps:

1. Initialize pose parameters.
2. For each iteration:
   (a) Project 3D surface onto the image plane.
   (b) Estimate the PDFs of the foreground and background regions, defined by the object's silhouette.
   (c) Calculate the gradient of the segmentation energy functional in terms of the pose parameters.
   (d) Propagate pose parameters in the direction of the optimal energy.

The PWP3D algorithm of Prisacariu and Reid (2012) follows an approach similar to Dambreville et al. (2010), while simplifying the calculations using level set functions. In contrast to Dambreville et al. (2010), who formulate the energy function using of two separate surface integrals over the foreground and the background, Prisacariu and Reid (2012) formulate the energy functions in terms of to sums over, using a Heaviside function evaluated over the embedding function, used to delineate each region. This step simplifies the energy gradient with respect to the pose parameters—the integral over the 3D occluding curve, is replaced with a sum over the 2D contour. Additionally, Prisacariu and Reid (2012) follow Bibby and Reid (2008), and define appearance models using posterior probability functions, instead of likelihood functions used by Dambreville et al. (2010). Bibby and Reid (2008) show that appearance models which rely on posterior probability functions are advantageous over likelihood functions. The PWP3D algorithm achieves similar results to those of the integrated approach suggested by Schmaltz et al. (2009), while running at real-time.

However, as we showed in Fig. 2 the global appearance models may be insufficient in capturing spatial variations in the foreground and background. Hence, a more sophisticated model which takes into account local variations is required. Thus, we suggest to apply ideas from local active contours (Lankton and Tannenbaum 2008) in order derive appearance models which capture the spatial variation in the foreground and background regions.

1.4 Localized 2D Image Segmentation

The idea of localizing segmentation calculations has been proposed in the past in different contexts. Rosenhahn et al. (2007) model the regions using varying local Gaussian probabilities. For each pixel they define a small window which is used to estimate the mean and standard deviation.

Schmaltz et al. (2009) concentrate on a free form surface consisting of rigid parts interconnected by predefined joints. Each part has its own appearance model, and the background is separated into multiple sub-regions, modeled using mixture models. They assume the background is static or slowly varying. They propose two algorithms for segmenting the background— K Means, which requires knowing the model order, or a level set algorithm which optimizes the number of regions. Assigning localized region models to different parts, could indeed have many advantages, when such a division is known a-priori. However, selecting the correct model order for the background segmentation could be a difficult task.

Horbert et al. (2011) combines the Implicit Shape Model Leibe et al. (2004) with a localized version of the posterior pixel-wise appearance models shown by Bibby and Reid (2008), focused on segmentation and tracking of pedestrians. This work attempts to capture the spatial variability using two appearance models for the foreground, one for the upper body parts and the second lower body parts, and two appearance models for the background. While separating the object into two regions seems like an adequate approach, modeling the background in terms of two appearance models may be insufficient in cluttered scenes.

Lankton and Tannenbaum (2008) present a framework for localizing active contours, i.e., propagating the active contour based on local region appearance models. For each point along the curve a local region is defined, and is split into a part interior and a part exterior to the curve. A local energy measuring the segmentation match between the two regions is defined. Each point is propagated in the direction that maximizes the segmentation, independently from other local decisions. This approach is very suitable to this segmentation problem as it addresses the spatial variation both in the foreground and in the background, without assuming any prior knowledge of the foreground and background. In contrast to Rosenhahn et al. (2007), where localized Gaussian distributions are defined, no assumptions are made regarding the probability functions. A key issue when using local region statistics is the size of the local regions. While Gaussian models are sufficient for very small regions, as the region sizes increase Gaussian models become insufficient and a more complex model must be considered.

1.5 Contribution

We propose a framework for simultaneous 3D pose estimation and image segmentation using local region statistics, instead of the global region statistics used in standard formulation. Local region statistics are capable of capturing spatial variation in image statistics. Thus we improve the 3D pose estimation in scenes containing heterogeneous objects or cluttered backgrounds.

We present the framework on the basis of the PWP3D algorithm (Prisacariu and Reid 2012), however it can be applied to other global region based methods, e.g., Dambreville et al. (2010).

We define a local energy function, which measures the segmentation quality within a local region. We fuse together the local energy functions into a single energy function and optimize it with respect to the pose parameters.

Finally, we present extensive experiments performed using the ACCV database (Hinterstoisser et al. 2012) comparing our algorithm's performance with the PWP3D algorithm (Prisacariu and Reid 2012). Pose estimation algorithms, such as PWP3D, may be applied as a first stage in more advanced algorithms (e.g., Dame et al. 2013). Enriching the basic algorithm by considering non homogeneous objects directly enriches each such advanced algorithm.

Structure of the paper—in Sect. 2 we present our approach to local region based pose estimation. Next, in Sect. 3 we discuss local region size selection. In Sect. 4 we discuss the implementation details. In Sect. 5 we present our results compared with the PWP3D algorithm. In Sect. 6 we make concluding remarks and give direction to future research.

2 Proposed Approach

In this section we present our proposed framework for extending the PWP3D algorithm (Prisacariu and Reid 2012) using local region statistics, instead of global region statistics. This section is divided into three parts—In Sect. 2.1 we present the formulation of the problem. In Sect. 2.2 we review the main steps of the PWP3D algorithm. In Sect. 2.3 we present our localized extension, highlighting the differences between our algorithm and the PWP3D.

We assume we are given a 3D surface model of an object located in an input image from a known calibrated camera.

Our objective is to find the transformation parameters that map the object's model and the object in the image.

Our algorithm relies an initialization of the pose parameters, and iteratively propagates the pose parameters using the gradients with respect to the pose parameters. The outline of our algorithm may be described using the following steps:

```
1. Initialize pose parameters. $\lambda = \lambda_0$
2. For each iteration:
   (a) Apply 3D transformation to object.
   (b) Project 3D model on to image plane.
   (c) For each local region:
       (i) Estimate local region statistics.
       (ii) Calculate local energy gradient with respect to pose parameters, $\nabla E_n$.
   (d) Fuse local region gradients, $\nabla E = f(\nabla E_n)$.
   (e) Find optimal step size, $s$.
   (f) Update pose parameters $\lambda = \lambda - s\nabla E$.
```

Where f is a function which fuses the local gradients, s is the step size and λ is the pose parameters vector. These steps are explained in depth throughout this section. Steps 2(a)-2(b), applying the 3D transformation and projecting the object on to image, are illustrated using the driller model in Fig. 5. Step 2(c), estimating the local region statistics, is shown for two different local regions in Fig. 6 and in Fig. 8. Their corresponding local foreground probability functions are shown in shown in Figs. 7b and 9b. Their corresponding local background probability functions are shown in shown in Figs. 7a and 9a. Each local region is affected by different elements—the background of the first local region is strongly affected by the blue bench vise, whereas the background of the second local region is strongly affected by the red ape model. These examples demonstrate the problem of describing the background and foreground regions using single appearance models when there is a strong spatial variation. This variation in statistical properties within each one of the regions makes it unreasonable to use a single appearance model to describe the entire region.

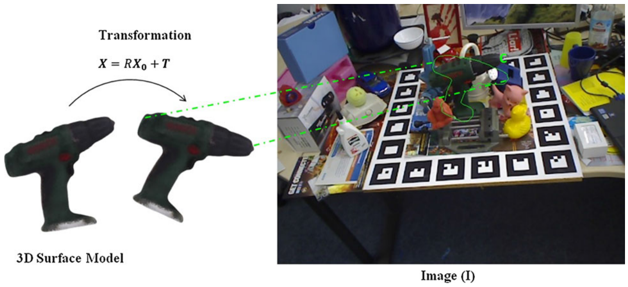


Fig. 5 Driller object projection onto the image plane

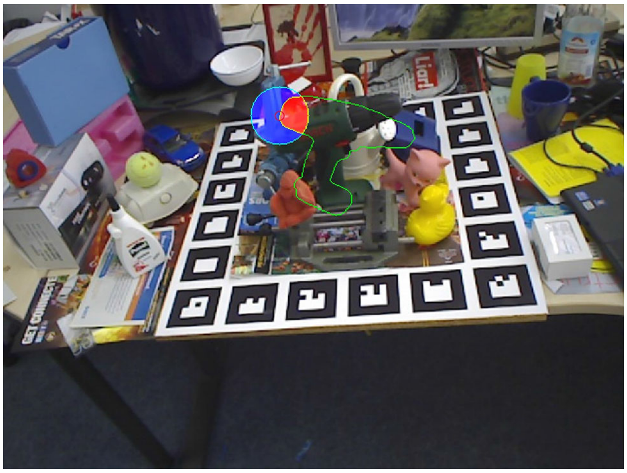
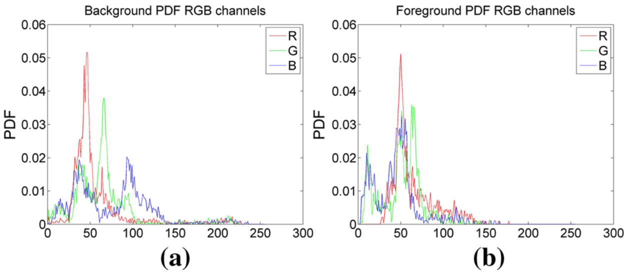


Fig. 6 Example of a local region extraction, divided into the local foreground (*red*) and local background (*blue*) (Color figure online)



(a) **(b)**

Fig. 7 Probability density functions of local region

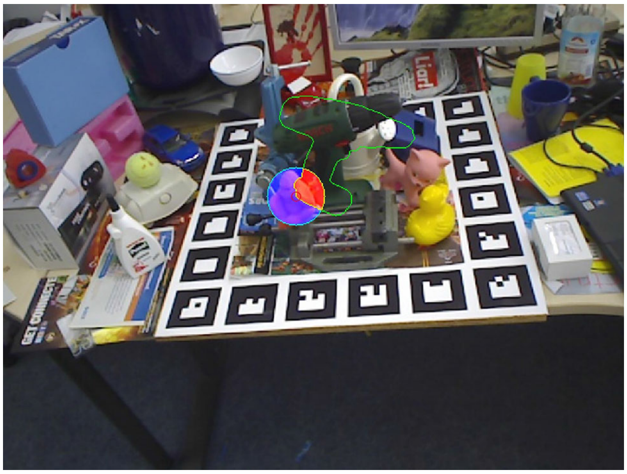


Fig. 8 Example of a local region extraction, divided into the local foreground (*red*) and local background (*blue*) (Color figure online)

2.1 Model

We begin by defining the rigid body transformation, which maps points from the object coordinate frame to the camera coordinate frame. A 3D point in the camera coordinate frame is denoted by $\boldsymbol{X} = [X, Y, Z]^T = \boldsymbol{R}\boldsymbol{X_0} + \boldsymbol{T} \in \mathbb{R}^3$. Where:

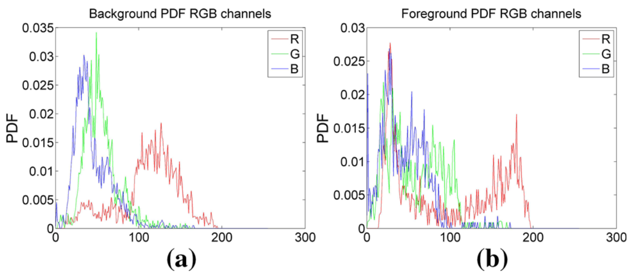


Fig. 9 Probability density functions of Local region

- $\boldsymbol{X_0} = [X_0, Y_0, Z_0]^T$ is the corresponding point in the object coordinate frame.
- $\boldsymbol{T} = [\lambda_1, \lambda_2, \lambda_3]^T$ denotes the translation vector in the x, y, z directions respectively.
- $\boldsymbol{R}$ is a rotation matrix represented in canonical exponential coordinates. It can be shown (Ma et al. 2003) that for any rotation matrix $R \in SO(3)$ (Lie group) there exists a $w \in \mathbb{R}^3$, $\|w\| = 1$ and $t \in \mathbb{R}^3$ such that:

$$\boldsymbol{R} = \exp(\hat{\boldsymbol{w}}t) \tag{1}$$

Where, $\hat{\boldsymbol{w}}$ is the skew symmetric matrix of the unit vector w. The unit vector w is the rotation axis and t is the rotation size in radians. We define the rotation vector as $[\lambda_4, \lambda_5, \lambda_6] = wt$, and a skew symmetric matrix:

$$\boldsymbol{\Lambda} = \begin{bmatrix} 0 & -\lambda_6 & \lambda_5 \\ \lambda_6 & 0 & -\lambda_4 \\ -\lambda_5 & \lambda_4 & 0 \end{bmatrix} \tag{2}$$

The rotation matrix is given by :

$$\boldsymbol{R} = \exp(\boldsymbol{\Lambda}) \tag{3}$$

The choice of exponential coordinates is merely due to their simplicity. Intrinsic camera parameters:

- $\left(f_x, f_y\right)$ are the focal distance in the x, y axes.
- (u_0, v_0) the principal points of the camera.

2.2 Global Region Based Pose Estimation

In this subsection we review the PWP3D global region based pose estimation framework developed by Prisacariu and Reid (2012), we reference the relevant equation numbers from the original work. In the following subsection present our extension of it to local region based 2D-3D pose estimation. Table 1 defines the notation that will be used in the context of global region framework: In Fig. 10 we illustrate the problem in terms of global regions properties.

Prisacariu and Reid (2012) showed that assuming pixel-wise independence the posterior probability of the shape of the contour, given the image data (Eq. 4):

Table 1 Global region notation

I	Input image
$\boldsymbol{x}$	2D pixel location in image. $\boldsymbol{x} = [u, v]$
$\boldsymbol{y}$	Pixel value ; $I(\boldsymbol{x}) = \boldsymbol{y}$
Ω	Image domain
Ω_f, Ω_b	Global foreground and background regions.
$W(\boldsymbol{x}, p)$	Warp with parameters p
C	Contour segmenting foreground from background
$\Phi(\boldsymbol{x})$	Shape kernel, level set embedding function
$P(\boldsymbol{y} \mid M_f)$	PDF of a pixel values $\boldsymbol{y}$, belonging to the foreground
$P(\boldsymbol{y} \mid M_b)$	PDF of a pixel values $\boldsymbol{y}$, belonging to the background
$H_\epsilon(z)$	Smoothed Heaviside function
$\delta_\epsilon(z)$	Smoothed delta function
Z	Photometric variable domain

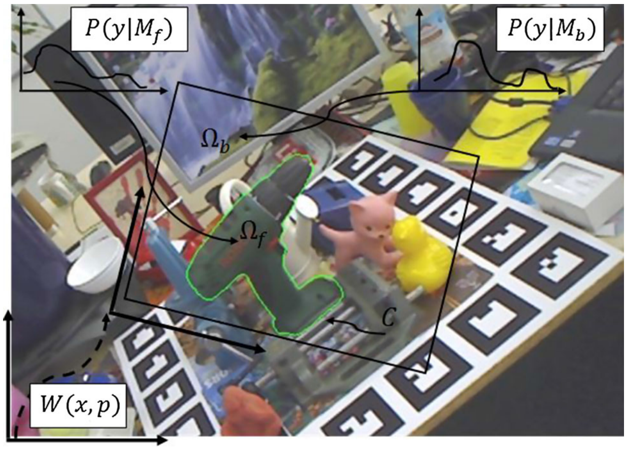


Fig. 10 Global region model

$$\begin{aligned} P(\Phi \mid I) = \prod_{\boldsymbol{x}\in\Omega} \big[& P_f(\boldsymbol{y}) H_e(\Phi(\boldsymbol{x})) \\ & + P_b(\boldsymbol{y})(1 - H_e(\Phi(\boldsymbol{x})))\big] \end{aligned} \tag{4}$$

Equation 4 describes the probability of the shape kernel Φ, defined by the constant and known 3D model, and the unknown pose parameters given the image data. Hence, it can be thought of as the posterior pose parameters probability given the image data. Where (Eqs. 7, 8):

$$P_f(\boldsymbol{y}) = \frac{P\left(\boldsymbol{y} \mid M_f\right)\eta_f}{P\left(\boldsymbol{y} \mid M_f\right)\eta_f + P(\boldsymbol{y} \mid M_b)\eta_b}$$
$$P_b(\boldsymbol{y}) = \frac{P(\boldsymbol{y} \mid M_b)\eta_b}{P\left(\boldsymbol{y} \mid M_f\right)\eta_f + P(\boldsymbol{y} \mid M_b)\eta_b}$$
$$\eta_f = \sum_{\boldsymbol{x}\in\Omega} H_e(\Phi(\boldsymbol{x})), \quad \eta_b = \sum_{\boldsymbol{x}\in\Omega} [1 - H_e(\Phi(\boldsymbol{x}))]$$

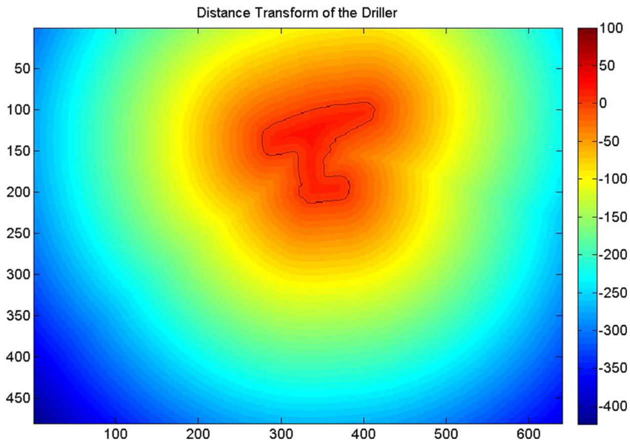


Fig. 11 Distance transform applied to driller

Φ is the distance transform defined as:

$$\Phi(\boldsymbol{x}) = \begin{cases} -d & \text{for} \quad \boldsymbol{x} \quad \text{inside the silhouette} \\ d & \text{for} \quad \boldsymbol{x} \quad \text{outside the silhouette} \end{cases}$$

Where d is the shortest distance between the pixel location $\boldsymbol{x}$ and the object's 2D contour. An example of a distance transform applied to the driller model is presented in Fig. 11, along with the 2D contour. The color bar to the side indicates the distance transform value, i.e., the signed minimal distance to object's contour from every pixel.

The energy function is defined as the negative log posterior probability (Eqs. 5, 6):

$$\begin{aligned} E &= -\log P(\Phi \mid I) \\ &= -\sum_{\boldsymbol{x}\in\Omega} \log\left[P_f H_e(\Phi) + P_b(1 - H_e(\Phi))\right] \end{aligned} \qquad (5)$$

The conditional probability functions may be estimated using a smoothed histogram, of the photometric variable chosen to perform the segmentation. We follow the choice of Prisacariu and Reid (2012) and use the photometric intensity of the RGB channels. A more sophisticated selection could be the usage of texture features, which could be important for textured objects (e.g., Zebra) as performed by Rosenhahn et al. (2007). Next, the energy function derivatives are calculated with respect to the pose parameters (Eqs. 11, 12):

$$\frac{\partial E}{\partial \lambda_i} = -\sum_{\boldsymbol{x}\in\Omega} \frac{P_f - P_b}{P_f H_e(\Phi) + P_b(1 - H_e(\Phi))} \frac{\partial H_e(\Phi)}{\partial \lambda_i} \qquad (6)$$

This equation is comprised of two components—the first (left hand part), relies on statistical properties estimated. The second, is a function of the objects geometry, independent of the statistical properties. Applying the chain rule we get:

$$\begin{aligned} \frac{\partial H_e(\Phi)}{\partial \lambda_i} &= \frac{\partial H_e(\Phi)}{\partial \Phi}\left[\frac{\partial \Phi}{\partial u}\frac{\partial u}{\partial \lambda_i} + \frac{\partial \Phi}{\partial v}\frac{\partial v}{\partial \lambda_i}\right] \\ &= \delta_e(\Phi)\left[\frac{\partial \Phi}{\partial u}\frac{\partial u}{\partial \lambda_i} + \frac{\partial \Phi}{\partial v}\frac{\partial v}{\partial \lambda_i}\right] \end{aligned} \qquad (7)$$

Substituting the camera model (Eqs. 13, 14):

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} \frac{X}{Z} f_x + u_0 \\ \frac{Y}{Z} f_y + v_0 \end{bmatrix}$$

$$\frac{\partial u}{\partial \lambda_i} = f_x \frac{\partial}{\partial \lambda_i}\frac{X}{Z} = f_x \frac{1}{Z^2}\left(Z\frac{\partial X}{\partial \lambda_i} - X\frac{\partial Z}{\partial \lambda_i}\right) \qquad (8)$$

$$\frac{\partial v}{\partial \lambda_i} = f_y \frac{\partial}{\partial \lambda_i}\frac{Y}{Z} = f_y \frac{1}{Z^2}\left(Z\frac{\partial Y}{\partial \lambda_i} - Y\frac{\partial Z}{\partial \lambda_i}\right) \qquad (9)$$

The differentials with respect to the translation parameters ($\lambda_1, \lambda_2, \lambda_3$) are given by:

$$\frac{\partial \boldsymbol{X}_j}{\partial \lambda_i} = \delta_{i,j} \qquad i, j = 1, 2, 3$$

The differentials with respect to the rotation parameters ($\lambda_4, \lambda_5, \lambda_6$) are given by:

$$\frac{\partial}{\partial \lambda_j}\boldsymbol{X} = \frac{\partial}{\partial \lambda_j}\boldsymbol{R}\begin{bmatrix} X_0 \\ Y_0 \\ Z_0 \end{bmatrix}$$

Where:

$$\frac{\partial}{\partial \lambda_j}\boldsymbol{R} = \exp(\boldsymbol{\Lambda})\begin{bmatrix} 0 & -\delta_{j,6} & \delta_{j,5} \\ \delta_{j,6} & 0 & -\delta_{j,4} \\ -\delta_{j,5} & \delta_{j,4} & 0 \end{bmatrix}, j = 4, 5, 6 \qquad (10)$$

Finally arriving at:

$$\frac{\partial}{\partial \lambda_j}\boldsymbol{X} = \boldsymbol{R}\begin{bmatrix} 0 & -\delta_{j,6} & \delta_{j,5} \\ \delta_{j,6} & 0 & -\delta_{j,4} \\ -\delta_{j,5} & \delta_{j,4} & 0 \end{bmatrix}\begin{bmatrix} X_0 \\ Y_0 \\ Z_0 \end{bmatrix}, j = 4, 5, 6 \qquad (11)$$

2.3 Local Region Based Pose Estimation

In this subsection we present our localized region based 3D pose estimation model. We define the notation that will be used in the context of local region framework in Table 2. The notation defined in the scope of global region segmentation remains unaffected.

We illustrate the local region parameters in Fig. 12. Following the approach used by Lankton and Tannenbaum

Table 2 Local region notation

$\boldsymbol{B}_n$	Binary mask of the n'th local region
d	Local region radius size in pixels
$P(\boldsymbol{y} \mid M_{f_n})$	PDF of a pixel values $\boldsymbol{y}$, belonging to the foreground in the n'th region
$P(\boldsymbol{y} \mid M_{b_n})$	PDF of a pixel values $\boldsymbol{y}$, belonging to the background in the n'th region.
Ω_n	n'th local region domain
Ω_{f_n}, Ω_{b_n}	Foreground and background regions in the n'th local region

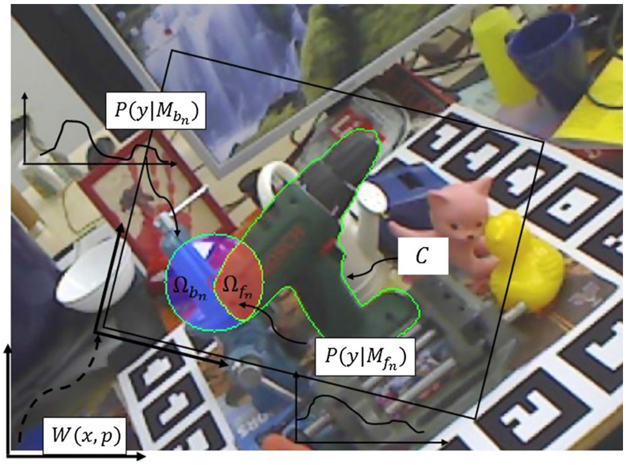


Fig. 12 Local region model

(2008) we define a characteristic function, $\boldsymbol{B}_n(\boldsymbol{x}_i)$, which masks local regions. The subscript n denotes the local region index:

$$\boldsymbol{B}_n(\boldsymbol{x}_i) = \begin{cases} 1 & \boldsymbol{x}_i \in \Omega_n \\ 0 & \boldsymbol{x}_i \notin \Omega_n \end{cases}$$

Specifically, we select $\boldsymbol{B}_n(\boldsymbol{x}_i, \boldsymbol{x}_c)$ as a circular binary mask centered at $\boldsymbol{x}_c$, with a radius size d:

$$\boldsymbol{B}_n(\boldsymbol{x}_i, \boldsymbol{x}_c) = \begin{cases} 1 & |\boldsymbol{x}_i - \boldsymbol{x}_c| < d \\ 0 & \text{else} \end{cases}$$

Next, we must determine how to divide the image domain into local regions. We recall from Sect. 2.3 that although the probability functions rely on statistical data collected over the entire image domain, the actual gradient calculation is performed using probability functions evaluated along the object's edge. This is depicted by the delta function multiplying the probability functions in Eq. (7). Hence, it is sufficient to define the local regions only along the contour. Specifically, we define a local regions around every point along the object's contour. We extend the generative posterior-pixelwise model of Bibby and Reid (2008), used by Prisacariu and Reid (2012) to define the global appearance models, to a *localized* pixel-wise posterior model. The full derivation is presented in Appendix. The expression we arrive at for the local energy of the n'th region is given by:

$$E_n = -\sum_{\boldsymbol{x}_i \in \Omega_n} \log \left[P_{f_n} H_\epsilon\left(\Phi(\boldsymbol{x}_i)\right) + P_{b_n}\left(1 - H_\epsilon\left(\Phi(\boldsymbol{x}_i)\right)\right) \right] \tag{12}$$

Equivalently, this may be written using the characteristic function, $\boldsymbol{B}_n(\boldsymbol{x}_i)$, as:

$$E_n = -\sum_{\boldsymbol{x}_i \in \Omega} \log \left[P_{f_n} H_\epsilon\left(\Phi(\boldsymbol{x}_i)\right) + P_{b_n}\left(1 - H_\epsilon\left(\Phi(\boldsymbol{x}_i)\right)\right) \right] \boldsymbol{B}_n(\boldsymbol{x}_i) \tag{13}$$

where P_{f_n}, P_{b_n} are the localized posterior probabilities of the n'th local region, which replace the global posterior probabilities of P_f, P_b. By replacing P_f, P_b with P_{f_n}, P_{b_n} we rely on the local statistical properties of each region, rather than global region statistics of the entire image. P_{f_n}, P_{b_n} are given by:

$$\begin{aligned} P_{f_n} &= \frac{P(\boldsymbol{y} \mid M_{f_n})}{\eta_{f_n} P(\boldsymbol{y} \mid M_{f_n}) + \eta_{b_n} P(\boldsymbol{y} \mid M_{b_n})} \\ P_{b_n} &= \frac{P(\boldsymbol{y} \mid M_{b_n})}{\eta_{f_n} P(\boldsymbol{y} \mid M_{f_n}) + \eta_{b_n} P(\boldsymbol{y} \mid M_{b_n})} \\ \eta_{f_n} &= \sum_{\boldsymbol{x}_i \in \Omega} B_n(\boldsymbol{x}_i) H_\epsilon(\Phi(\boldsymbol{x}_i)) \\ &= \sum_{\boldsymbol{x}_i \in \Omega_n} H_\epsilon(\Phi(\boldsymbol{x}_i)) \\ \eta_{b_n} &= \sum_{\boldsymbol{x}_i \in \Omega} B_n(\boldsymbol{x}_i)(1 - H_\epsilon(\Phi(\boldsymbol{x}_i))) \\ &= \sum_{\boldsymbol{x}_i \in \Omega_n} (1 - H_\epsilon(\Phi(\boldsymbol{x}_i))) \\ \eta_n &= \eta_{f_n} + \eta_{b_n} \end{aligned}$$

The energy function which fuses the N local regions is defined as:

$$E = \frac{1}{N} \sum_{n=1}^{N} E_n \tag{14}$$

$$E = -\frac{1}{N} \sum_{n=1}^{N} \sum_{\boldsymbol{x}_i \in \Omega} \log \left[P_{f_n} H_\epsilon\left(\Phi(\boldsymbol{x}_i)\right) + P_{b_n}\left(1 - H_\epsilon\left(\Phi(\boldsymbol{x}_i)\right)\right) \right] \boldsymbol{B}_n(\boldsymbol{x}_i) \tag{15}$$

This equation shows the crucial difference between our method and the PWP3D. The PWP3D energy, shown in

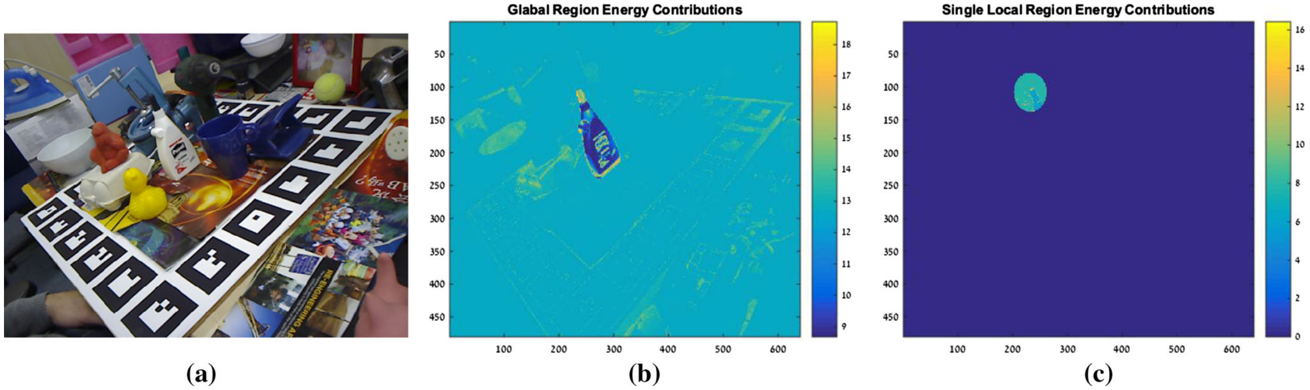


Fig. 13 Energy function evaluation over heterogeneous object

Eq. 2.2 relies on a single appearance model for the entire image domain, while our algorithm relies on N different local regions. In the PWP3D algorithm all the points considered in the summation rely on the same appearance model, while in our algorithm every point considered has its own appearance model, based on its local surroundings. The gradients of the energy function with respect to the pose parameters are given by:

$$\frac{\partial E}{\partial \lambda_i} = -\frac{1}{N}\sum_{n=1}^{N}\sum_{\boldsymbol{x}_i\in\Omega_n}\frac{P_{f_n}-P_{b_n}}{P_{f_n}H_e(\Phi)+P_{b_n}(1-H_e(\Phi))} \times \frac{\partial H_e(\Phi)}{\partial \lambda_i}B_n(\boldsymbol{x}_i) \qquad (16)$$

The term $\frac{\partial H_e(\Phi)}{\partial \lambda_i}$ depends strictly on the geometry of the object, they can be interpreted as the geometric differentials of the object with respect to the pose parameters. The term $\frac{P_{fn}-P_{b_n}}{P_{f_n}H_e(\Phi)+P_{b_n}(1-H_e(\Phi))}$ can be interpreted as the weight applied to the geometrical differentials, based on the statistical fit. In the PWP3D, Eq. 2.2, this element was determined based on the fit of the global appearance model to a given point. Whereas our extension, Eq. (16), weighs the geometrical differentials based on the fit of the local appearance model's fit at the given point.

We illustrate the impact of the localized energy function on the basis of the glue object from the ACCV 2012 database (Hinterstoisser et al. 2012). The image in Fig. 13a shows the glue object, which we consider as heterogeneous—its body is white, however its tip is black and it has black texture on the body. We evaluate the term $\log\left[P_f H_e(\Phi) + P_b(1-H_e(\Phi))\right]$ from Eq. 2.2 over the entire image, in Fig. 13b. This term is the contribution of each pixel in the foreground and background to the energy function. The figure illustrates the problem of using global appearance models—the black area adds a penalty on the energy function causing the object avoid such areas. Next, we evaluate the term $\log\left[P_{f_n}H_\epsilon(\Phi(\boldsymbol{x}_i)) + P_{b_n}(1-H_\epsilon(\Phi(\boldsymbol{x}_i)))\right]$ from Eq. 12 over a single local region in Fig. 13c. This figure shows that the penalty due to including the glue's black tip is considerably lower with respect to the global appearance models.

3 Local Region Size Selection

A key issue in the framework of local region segmentation is the selection of the region's sizes on which the local statistics will be estimated. We use circular local regions and determine their size by setting their radius size. By adjusting the regions' sizes we determine to what extant we use global or local region statistics:

1. As the region's sizes increases the local regions become more correlated, with less variation between regions, thus leaning towards global region statistics. When the regions sizes exceed the size of the image, they will be identical, arriving at the original global model.
2. As the radius's sizes decrease the local regions become less correlated, the variation between regions change more swiftly, allowing them to better capture the spatial variation.

The region size selection offers a basic trade off between robustness and the capability to capture the spatial variation—Using large sized regions, the region statistics will be more robust to the initialization of the object's pose. Changes in the objects pose will have a lower impact on the statistical models. However, as we demonstrated in Fig. 2 the large region was insufficient in capturing the varying foreground and background variation in statistical properties. Using small sized regions, the ability to capture variations in the region's statistics increases, as shown in Fig. 4a, b. The downfall of a small radius size is the loss of robust-

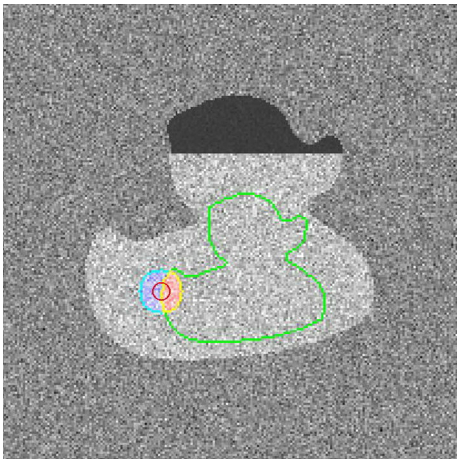
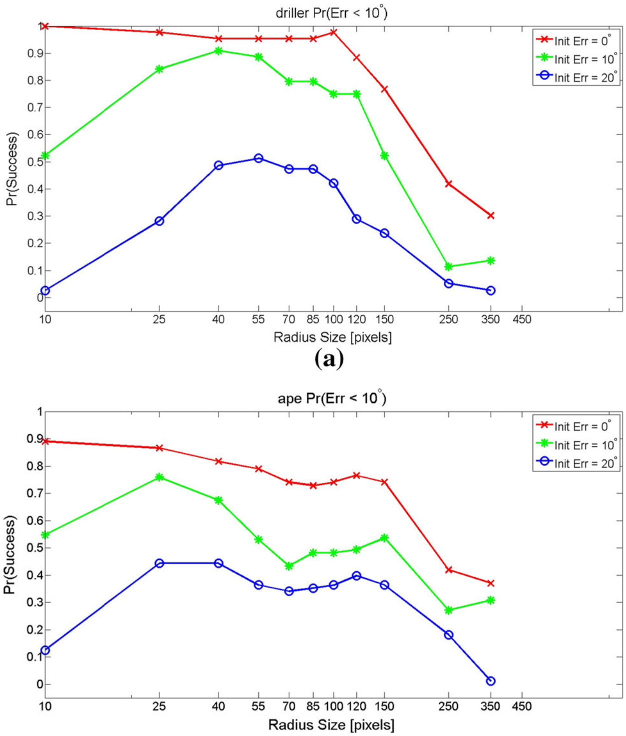


Fig. 14 Example of over-fitting due to small radius selected (radius = 10)

ness. Small changes in the objects pose may strongly affect the region's statistics to the extant of over-fitting. In this case the local foreground and local region' statistics will not capture the true statistical properties of the foreground and background but rather the statistics of the region. The algorithm will optimize the pose of the object using the statistics of the local regions, rather than the true foreground and background. Consider for example the duck object in Fig. 14. Due to a poor scale selection the local region, shown as a circle, contains only foreground statistics. In this case the algorithm is likely to keep the object in the same location, as the energy will indicate a good segmentation.

The radius selection is directly related to the well known trade off of model order selection. The model order is inversely proportional to the radius size—a small radius will result in many statistically independent regions, hence a high order model, whereas selecting a large radius will result in a higher correlation between the regions, restricting the model order. The trade-offs in region size is between descriptiveness, which increases with the model order, at the cost of robustness, which increases the model order decreases. Lankton and Tannenbaum (2008) studied the problem of radius size selection for localized active contour segmentation. They suggest selection of the parameter based on the scale of the object. A small object or a cluttered background require a small radius to correctly capture the variation between regions, whereas for a large object and a slowly varying background, a large radius is preferred. Their results are applicable to our problem as well, with a few subtle differences. In the active contours problem each point is free to move independently from the other points, whereas in our problem, the 3D model imposes a geometric constraint on the possible pose parameter propagation. This constraint is depicted in the gradient calculation,



(a)

(b)

Fig. 15 Probability for a correct pose estimation as a function of radius size, for various rotation error sizes

Eq. (16). In this equation a summation over the local regions is performed, hence the power of a single local region is limited.

We explored the impact of the radius size by examining the probability of an error as a function of radius size for various initial rotation errors. We defined a correct pose as a final error of less than 10 degrees. We performed this experiment using two objects—one a heterogeneous driller object, and the second a homogeneous ape object. The results of the experiments are presented in Fig. 15. A strong improvement is shown for the non-homogeneous driller, where the global model is insufficient, and little improvement for the homogeneous ape model, where the global model is expected to be sufficient.

The results demonstrate several issues discussed earlier:

1. Robustness—The performance of the algorithm is relatively stable for a wide range of radius sizes, from a radius of 40–120 pixels.
2. Over-fit—Selecting a very small radius (10 pixels) results in an over-fit, the performance is very good for small initial errors, and degrades as the initial error increases.
3. Global model insufficiency—for large radius sizes the performance is severely impacted. This is due to the insufficient model described earlier.

4 Implementation Details

4.1 Local Region Dilution

Defining a set of local regions on which the statistical calculations are performed has serious run time implications. In order to reduce run time we define a parameter Dilution Factor as the distance between local region where the local statistics are actually performed. In the intermediate local regions the statistics are estimated by performing a linear interpolation between the nearest regions. In our work we selected a dilution factor d = 0.05R.

4.2 Run Time

An important parameter in determining the feasibility of applying an algorithm in a realistic systems is its run time. The global PWP3D presented run time at the order of several milliseconds by parallelizing the computations using the CUDA framework on a Geforce video card. Applying the localized algorithm requires computation of the statistical properties on multiple local regions, in contrast to the global algorithm where the computation is performed in a single region. The run time required for histogram calculation may increase by a factor of $O(N)$, N being the number of regions, due to the independent calculation required for each region. In practice this run time is expected to be considerably lower, as the region size on which each local histogram is performed is considerably smaller. The total run time is a function of many parameters—the 3D model complexity, code efficiency, hardware, etc. In order to compare the run time of the two algorithms we measured the average run time of each iteration for various local region sizes. We performed this experiment with two objects—the driller model and the ape model. We implemented both algorithms using the MATLAB Parallel Computing Toolbox MathWorks (2014) applied to CPU calculations, without GPU optimization. The run time ratio we arrived at was between 400 for a very small radius and 10 for radius sizes of approximately 20 pixels and above. This dependency on the radius size is a result of our selection of the dilution parameter as a factor of the radius size—as the radius size increases the number of actual calculations performed reduces.

Despite the increase in run time relative to the PWP3D, our algorithm's run time may be significantly improved by parallelizing the histogram calculation within each local region. Additionally, using local appearance models requires estimating the regions statistics only over a smaller region of the image, rather than over the entire image. By using appropriate hardware (e.g., multi-processor GPU) and software language we estimate our algorithm will achieve similar run time performance as the PWP3D.

Recently, Prisacariu et al. (2013) presented a framework for joint 3D tracking and reconstruction on a mobile phone. The framework used for tracking is based on the global PWP3D of Prisacariu and Reid (2012), however by applying several optimizations the are able to present a mobile phone applicable algorithm. The run time optimization is applied to the three most time costly procedures—(i) Rendering—the projection of the 3D object onto the image is performed using a hierarchical binary rendering scheme. (ii) Efficient calculation of the Signed Distance Transform (SDF) derivatives—the computation of the (SDF) Φ derivatives is approximated only in a narrow band around the objects edge instead of over the entire object. (iii) Optimization—they apply Levenberg-Marquardt algorithm to find the optimal pose parameters from the energy function gradients. These steps are applicable to our framework as well in order to improve the run time.

4.3 Conditional Probabilities Estimation

The conditional PDFs are estimated by calculating the histograms (256 bins) of each color space and smoothing them using a Gaussian kernel. For simplicity we assumed the RGB channels are independent, therefor:

$$P\left(\mathbf{y}_{RGB} \mid M\right) = P\left(y_R \mid M\right) P\left(y_G \mid M\right) P\left(y_B \mid M\right) \quad (17)$$

However, using more realistic color models could be considered for better segmentation.

4.4 Optimization

In order to find the pose parameters which minimize the energy function we apply a simple first order gradient based optimization scheme. This iterative scheme requires finding the optimal step size selection in every iteration. We start by normalizing the rotation gradient and the translation gradient each to unit pixel and unit rotation size. We restrict our step sizes such that all translation components share the same step size, and all rotation components share another step size. This is required in order to keep the gradient of the rotation vector in the correct direction. The result is a 2D search for optimal rotation and translation step sizes, where the optimal value selected is the one which achieves the minimal energy value. We employ a coarse to fine search method—we reduce or increase the search region as the optimal step size decreases or increases.

5 Results

In order to demonstrate the strengths of our method, we performed a series of experiments, comparing the basin of

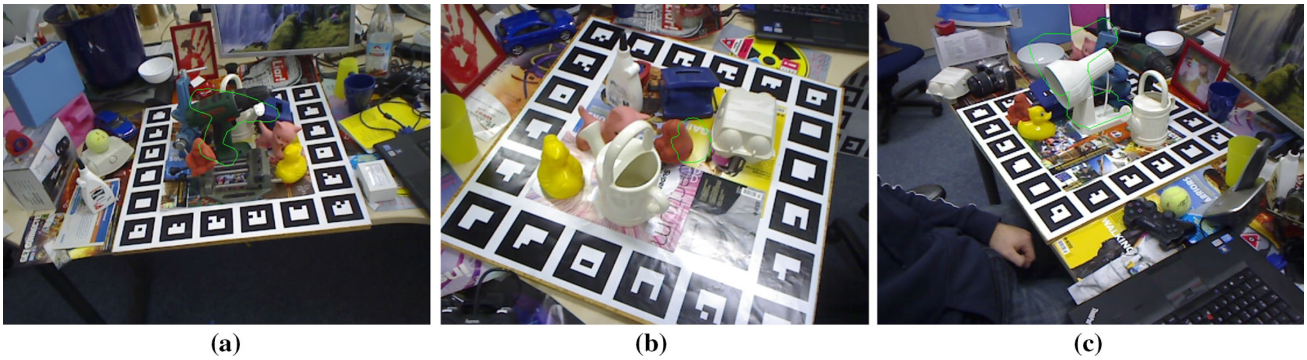


Fig. 16 Illustration of the maximal offsets of the rotation, translation and scale parameters where the algorithm may estimate the correct pose

attraction of our algorithm (labeled as local PWP3D) and the original PWP3D algorithm (Prisacariu and Reid 2012) (labeled as global PWP3D). The basin of attraction is defined in dynamical systems as the set of initial conditions leading to long-term behavior. In our case, it is used to measure the range of initial pose errors (angular or transnational) for which the algorithm converges to the correct pose. Based on the region size, a sampling scheme of initial guesses can be constructed in order to reliably estimate the pose of an object. We performed these experiments using the ACCV 2012 database of Hinterstoisser et al. (2012). The database comprises 15 different objects, with different levels of heterogeneity in a highly cluttered background. We selected a representative subset of objects (Fig. 17) which we divided into homogeneous and heterogeneous, and performed the following experiments:

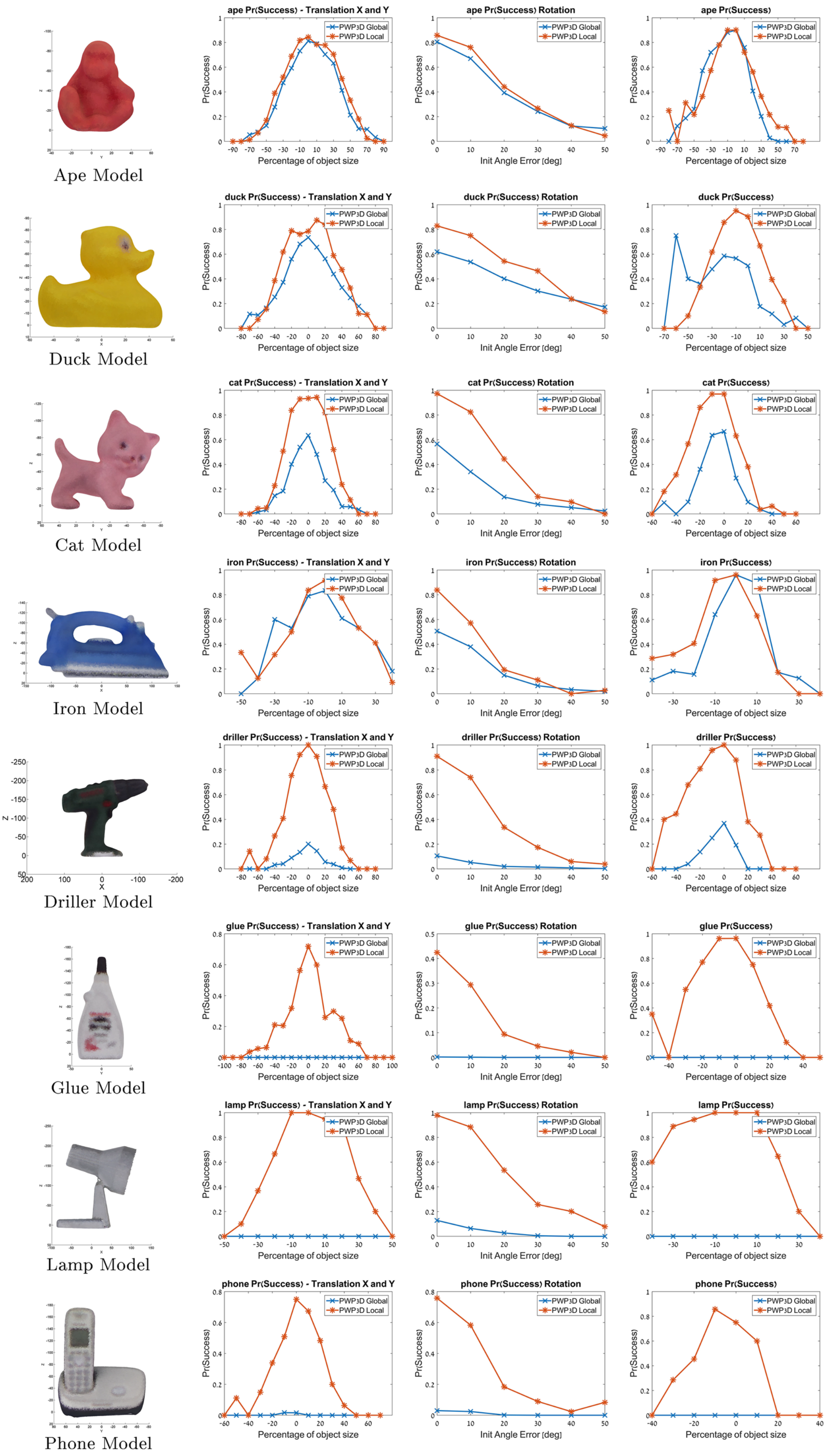
(a) Rotation angle basin of attraction—in this experiment, the object's pose was initialized to some erroneous rotation around it's center of mass and measured the probability of convergence to the correct pose. The axis of rotation is selected randomly, hence the rotation value is unsigned. The results of this experiment for each object are shown in the second column of Fig. 17. We illustrate the edge of the convergence region for the driller object (30 degrees) in Fig. 16a.
(b) Translation parameters basin of attraction in the X and Y directions—we performed a similar experiment to compare the translation parameters basin of attraction. In this experiment we initialized the object's pose to some erroneous translation in the x and y directions and measured the probability of estimating the correct pose.The results of this experiment for each object are shown in the third column of Fig. 17. We illustrate the edge of the convergence region for the ape object (60 % degrees) in Fig. 16b.
(c) Scale parameter basin of attraction by setting the Z offset—in this experiment we measured the basin of attraction of the scale parameter by selecting offsets in the Z axis and measuring the probability of estimating the correct pose. The results of this experiment for each object are shown in the fourth column of Fig. 17. We illustrate the edge of the convergence region for the lamp object (40 % degrees) in Fig. 16c.

The success criteria defined in these experiments was a rotation error of at most 10 degrees and 10% of object size in translation. In all experiments we set the local region's radius size to 30 pixels for all objects, despite their variability in size and shape. The consistent results across various objects achieved, despite the non optimal radius selected, indicates good robustness to radius size. The results for all three experiments show similar trends: For fairly homogeneous objects (Ape, Cat, Duck, Iron) the performance of the local PWP3D and the global PWP3D is very similar—the global appearance models are sufficient in order to describe the object. However, for heterogeneous models (Driller, Phone, Lamp, Glue) our algorithm shows significant improvement. Some of the objects in the heterogeneous group are more heterogeneous (e.g., glue, phone, lamp) than others (driller) and thus a more severe degradation is observed. We emphasize that the differences in performance are independent of the optimization algorithm selected. This is depicted by the results at a rotation angle of 0 degrees. An incorrect pose estimated at an initial error of 0 degrees indicates a minimum which is below the energy level of the ground truth. Hence, using the global model for heterogeneous object the optimal segmentation does not correspond to the correct pose.

5.1 Impact of Local Region Radius Size on Rotation Angle Basin of Attraction

In this section we present results of the rotation angle basin of attraction for various radius sizes. The experiment is an extension basin of attraction of the rotation angle parameter, to various local region radius sizes. We performed this experiment on the homogeneous Ape model and heterogeneous Driller model. As discussed in Sect. 3 the radius size

Fig. 17 Performance analysis: probability of estimating the correct pose of each object. Homogeneous objects: Ape, Duck, Cat, Iron; heterogeneous objects: Driller, Glue, Lamp, Phone



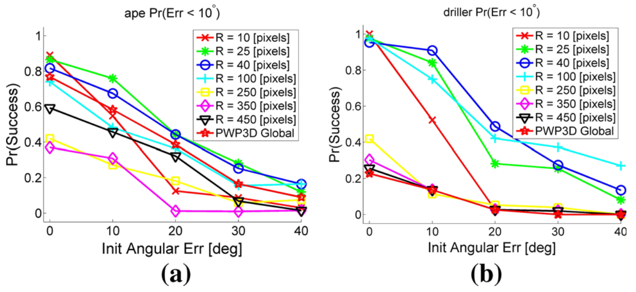


Fig. 18 Rotation angle basin of attraction for various local region radius sizes. R is the radius size in pixels of the local region

selection is a tradeoff problem. Selection of too large a radius doesn't allow to properly capture the statistics, whereas selection of too small a radius leads to over-fitting. This is apparent in the results presented in Fig. 18.

- Ape—The basin of attraction of the ape model remains unaffected for most radius size selections, with the exception of the radius size of 10 pixels. This behavior is due to the homogeneity which does not require a more complex model. The exceptional case, where a radius size of 10 pixels is selected shows good behavior for small initial errors and decreases as the initial error grows. This type of behavior is typical of over-fitting.
- Driller—For this object we observe three different behaviors:
  1. $R = 10$—In this case we observe the over-fitting once again.
  2. $10 < R < 100$—For this range of radiuses the localized algorithm behaves fairly well.
  3. $R > 100$—For this range of radiuses, which includes global PWP3D, the performance degrades, as the model cannot capture the spatial variability of the driller object.

6 Conclusions

In this manuscript we have presented a novel framework for simultaneously estimating the 3D pose of an object and 2D image segmenting using a localized region based approach. Inspired by ideas shown for local active contours, we extend the PWP3D algorithm such that the segmentation is performed using local region statistics rather than global region statistics. This crucial difference allows us to extend the PWP3D algorithm to a new domain of objects, which are not homogeneous. We formulate our extension by defining multiple local energy functions, measuring the segmentation within each local region, and fusing them into a single energy function measuring the overall segmentation quality. Next we derive the gradients of the energy function with respect to the pose parameters. We experimented with our localized region based framework, comparing it with the recent PWP3D and showed a dramatic improvement for heterogeneous objects. Furthermore we show a considerable improvement in the performance for a wide range of radius sizes selected for the local regions. The measured basin of attraction indicates our algorithm could be suitable for a pose estimation scheme, and not only for 3D tracking, where a narrow basin of attraction is required due to small frame by frame variation.

Appendix: Localized PWP3D Energy Functional

In this appendix we present the formulation of our energy function, using local region statistics, in contrast to the energy function shown by Bibby and Reid (2008), which uses global region statistics.

The key difference between the two energy functions lies in the statistical assumption, used in order to fuse the information from different pixels in the image. Bibby and Reid (2008) apply a logarithmic opinion pool (LOP), equivalent to assuming that the pixels within each global region may be treated as statistically independent and identically distributed. The assumption that all pixels, within each global region, are identically distributed leads to global appearance models. In contrast, we assume the assumption that pixels are identically distributed and independently distributed is valid only within small regions. Our assumption leads to the derivation of local appearance models, used to describe small parts of the image. Instead of describing the foreground and background regions using global appearance models, we use multiple *local* appearance models. Consequently, the energy function used to measure the segmentation quality relies on multiple local regions, rather than global appearance models.

For ease of comparison with the global region model, we briefly review the main stages used in the global model derivation in Appendix 1. In Appendix 2 we derive our energy function which relies on local region statistics.

Following the notation of Prisacariu and Reid (2012) we define the energy function as:

$$E = -log\,(P(\Phi \mid I)) \tag{18}$$

However, Φ is a direct function of the known 3D model and the pose parameters. Thus, it is simply the posterior probability of the pose parameters given the image data.

Appendix 1: Global Region Formulation

Bibby and Reid (2008) show that the joint distribution for a single pixel is given by (eq (1) in ref):

$$\begin{aligned}
&P(\boldsymbol{x}, \boldsymbol{y}, \Phi, p, M)\\
&\quad = P(\boldsymbol{x}, \boldsymbol{y} \mid \Phi, p, M)P(\Phi, p, M))\\
&\quad = P(\boldsymbol{x} \mid \Phi, p, M)P(\boldsymbol{y} \mid M)P(\Phi)P(p)P(M)\\
&\quad = P(\boldsymbol{x} \mid \Phi, p, M)P(\boldsymbol{y}, M)P(\Phi)P(p) \qquad (19)
\end{aligned}$$

Where:

$$M \in \{M_f, M_b\} \qquad (20)$$

Dividing by $P(\boldsymbol{y}) = \sum_{j=\{f,b\}} p(\boldsymbol{y} \mid M_j)P(M_j)$ leads to (eq. (2) in ref):

$$P(\boldsymbol{x}, \Phi, p, M \mid \boldsymbol{y}) = P(\boldsymbol{x} \mid \Phi, p, M)P(M \mid \boldsymbol{y})P(\Phi)P(p)$$

Next, they apply Bayes rule (eq(3) in ref):

$$P(M_j \mid \boldsymbol{y}) = \frac{P(\boldsymbol{y} \mid M_j)P(M_j)}{\sum_{\{i=f,b\}} P(\boldsymbol{y} \mid M_i)P(M_i)} \quad j = \{f, b\} \qquad (21)$$

and marginalize over the models M_j :

$$\begin{aligned}
&P(\boldsymbol{x}, \Phi, p \mid \boldsymbol{y})\\
&\quad = P(\Phi)P(p) \sum_{j=f,b} P(\boldsymbol{x} \mid \Phi, p, M_j)P(M_j \mid \boldsymbol{y}) \qquad (22)
\end{aligned}$$

Dividing by $P(\boldsymbol{x})$ leads to (eq (4) in ref):

$$\begin{aligned}
P(\Phi, p \mid \boldsymbol{y}, \boldsymbol{x}) = P(\Phi)P(p)\frac{1}{P(\boldsymbol{x})}\\
\times \sum_{j=f,b} P(\boldsymbol{x} \mid \Phi, p, M_j)P(M_j \mid \boldsymbol{y})
\end{aligned}$$

The term $1/P(\boldsymbol{x})$ can be dropped as it is constant for all pixel locations. Finally, arriving at the probability of the shape and the location p given pixel $\{\boldsymbol{x}, \boldsymbol{y}\}$

Next, a logarithmic opinion pool is applied in order to fuse together the pixel-wise posteriors:

$$\begin{aligned}
P(\Phi, p \mid \Omega) = P(\Phi)P(p) \prod_{i=1}^{K}\\
\times \sum_{j=f,b} P(\boldsymbol{x}_i \mid \Phi, p, M_j)P(M_j \mid \boldsymbol{y}_i) \qquad (23)
\end{aligned}$$

In this step lies the main difference between our approach—we do not assume that all pixels are identically distributed, but only within a local region.

Next, the term $P(p)$, which is constant for all pixels, and the term $P(\Phi)$, which is unnecessary as Φ is known from the 3D surface model, are dropped.

The first term, $P(\boldsymbol{x}_i \mid \Phi, p, M_j)$, may be written as:

$$P(\boldsymbol{x}_i \mid \Phi, p, M_j) = \begin{cases} \frac{H_\epsilon(\Phi(\boldsymbol{x}_i))}{\eta_f} & j = f\\ \frac{1-H_\epsilon((\Phi(\boldsymbol{x}_i))}{\eta_b} & j = b \end{cases}$$

The second term, $P(M_j \mid \boldsymbol{y})$, can be calculated using Eq. (21). The term $P(\boldsymbol{y} \mid M_j)$ is simply the likelihood function calculated based on the statistics of each region.

$$\begin{aligned}
&P(M_j) = \begin{cases} \frac{\eta_f}{\eta} & j = f\\ \frac{\eta_b}{\eta} & j = b \end{cases}\\
&\eta_f = \sum_{i=1}^{K} H_\epsilon(\Phi(\boldsymbol{x}_i)),\ \ \eta_b = \sum_{i=1}^{K}(1 - H_\epsilon(\Phi(\boldsymbol{x}_i))),\\
&\eta = \eta_f + \eta_b = K,
\end{aligned}$$

Summing to :

$$\begin{aligned}
P(\Phi, p \mid \Omega) = \prod_{i=1}^{K} \big[P_f(\boldsymbol{y_i})H_\epsilon(\Phi(\boldsymbol{x}_i) + P_b(\boldsymbol{y}_i)\\
(1 - H_\epsilon\,(\Phi\,(\boldsymbol{x}_i)))\big]
\end{aligned}$$

Where:

$$\begin{aligned}
P_f &= \frac{P(\boldsymbol{y} \mid M_f)}{\eta_f P(\boldsymbol{y} \mid M_f) + \eta_b P(\boldsymbol{y} \mid M_b)}\\
P_b &= \frac{P(\boldsymbol{y} \mid M_b)}{\eta_f P(\boldsymbol{y} \mid M_f) + \eta_b P(\boldsymbol{y} \mid M_b)}
\end{aligned}$$

The energy is defined as minus the log-posterior probability:

$$\begin{aligned}
E &= -\log P\,(\Phi \mid I) \qquad (24)\\
&= -\sum_{\boldsymbol{x}\in\Omega} \log\big[P_f H_e\,(\Phi) + P_b\,(1 - H_e\,(\Phi))\big] \qquad (25)
\end{aligned}$$

Appendix 2: Local Region Framework

We now present the derivation of our energy function which relies on local statistics. The main difference lies in Eq. (23). Instead of assuming that the pixels' distributions are identical and independent we apply this assumption only within local regions, i.e., we assume that within each local region they are distributed identically and independently. Local regions are assumed to be independent one from another. We begin by deriving the framework for a single local region. This derivation is essentially equivalent to the derivation of the global model, as it follow the same assumptions. Next, we fuse together the local region statistics and arrive at the localized energy function.

Appendix 2(a) Posterior Probability of a Single Local Region

We begin by defining the posterior probability of a local region as $P(\Phi, p \mid \Omega_n)$. Within each local region we assume that the pixels may be assumed to be treated as independent and identically distributed. Hence, for the calculation of $P(\Phi, p \mid \Omega_n)$ we apply the global region model developed in the previous section. For all $\boldsymbol{x} \in \Omega_n$ the joint probability can be written as:

$$\begin{aligned} &P(\boldsymbol{x}, \boldsymbol{y}, \Phi, p, M_n) \\ &\quad = P(\boldsymbol{x}, \boldsymbol{y} \mid \Phi, p, M_n) P(\Phi, p, M_n)) \\ &\quad = P(\boldsymbol{x} \mid \Phi, p, M_n) P(\boldsymbol{y} \mid M_n) P(\Phi) P(p) P(M_n) \\ &\quad = P(\boldsymbol{x} \mid \Phi, p, M_n) P(\boldsymbol{y}, M_n) P(\Phi) P(p) \end{aligned}$$

The posterior model probability given a pixels value becomes:

$$P(M_{n_j} \mid \boldsymbol{y}) = \frac{P(\boldsymbol{y} \mid M_{j_n}) P(M_{n_j})}{\sum_{i=f,b} P(\boldsymbol{y} \mid M_{n_i}) P(M_{n_i})} \qquad j = \{f, b\} \ , \boldsymbol{x}_i \in \Omega_n \tag{26}$$

The joint probability of the pixel location, embedding function and warping parameter becomes:

$$\begin{aligned} &P(\boldsymbol{x}, \Phi, p, M_n \mid \boldsymbol{y}) \\ &\quad = P(\boldsymbol{x} \mid \Phi, p, M_n) P(M_n \mid \boldsymbol{y}) P(\Phi) P(p) \ \boldsymbol{x}_i \in \Omega_n \end{aligned}$$

Applying the logarithmic opinion pool across the local region pixels we obtain:

$$\begin{aligned} &P(\Phi, p \mid \Omega_n) \\ &\quad = \prod_{\boldsymbol{x}_i \in \Omega_n} \sum_{j=f,b} P(\boldsymbol{x} \mid \Phi, p, M_{n_j}) P(M_{n_j} \mid \boldsymbol{y}) P(\Phi) P(p) \end{aligned} \tag{27}$$

We define a binary mask, which will facilitate in explicitly selecting the local region pixels:

$$\boldsymbol{B}_n(\boldsymbol{x}_i) = \begin{cases} 1 & \boldsymbol{x}_i \in \Omega_n \\ 0 & \boldsymbol{x}_i \notin \Omega_n \end{cases}$$

$$\begin{aligned} P(\Phi, p \mid \Omega_n) = \prod_{i=1}^{K} \boldsymbol{B}_n(\boldsymbol{x}_i) \sum_{j=f,b} P\left(\boldsymbol{x} \mid \Phi, p, M_{n_j}\right) \\ \times P\left(M_{n_j} \mid \boldsymbol{y}\right) \cdot P(\Phi) P(p) \end{aligned} \tag{28}$$

Where K is the total number of pixels in the image. Once again we may omit $P(\Phi)$, $P(p)$, arriving at:

$$P(\Phi, p \mid \Omega_n) = \prod_{\boldsymbol{x}_i \in \Omega_n} \sum_{j=f,b} P(\boldsymbol{x} \mid \Phi, p, M_{n_j}) P(M_{n_j} \mid \boldsymbol{y}) \tag{29}$$

Where:

$$P(\boldsymbol{x}_i \mid \Phi, p, M_{n_j}) = \begin{cases} \frac{H_\epsilon(\Phi(\boldsymbol{x}_i))\boldsymbol{B}_n(\boldsymbol{x}_i)}{\eta_{fn}} & j = f \\ \frac{[1 - H_\epsilon(\Phi(\boldsymbol{x}_i))]\boldsymbol{B}_n(\boldsymbol{x}_i)}{\eta_{b_n}} & j = b \end{cases}$$

Applying Eq. (26):

$$P(M_{n_f} \mid \boldsymbol{y}) = \frac{P(\boldsymbol{y} \mid M_{n_f})}{\eta_{f_n} P(\boldsymbol{y} \mid M_{n_f}) + \eta_{b_n} P(\boldsymbol{y} \mid M_{n_b})} \equiv P_{f_n}$$

$$P(M_{n_b} \mid \boldsymbol{y}) = \frac{P(\boldsymbol{y} \mid M_{n_b})}{\eta_f P(\boldsymbol{y} \mid M_{n_f}) + \eta_b P(\boldsymbol{y} \mid M_{n_b})} \equiv P_{b_n}$$

Finally arriving at:

$$\begin{aligned} P(\Phi, p \mid \Omega_n) = \prod_{i=1}^{K_n} \big[& P_{f_n} H_\epsilon(\Phi(\boldsymbol{x}_i)) \boldsymbol{B}_n(\boldsymbol{x}_i) \\ & + P_{b_n} \left(1 - H_\epsilon(\Phi(\boldsymbol{x}_i))\right) \boldsymbol{B}_n(\boldsymbol{x}_i) \big] \end{aligned} \tag{30}$$

$$E_n = - \sum_{\boldsymbol{x}_i \in \Omega_n} \log \left[P_{f_n} H_\epsilon \left(\Phi(\boldsymbol{x}_i)\right) + P_{b_n} \left(1 - H_\epsilon \left(\Phi(\boldsymbol{x}_i)\right)\right) \right] \tag{31}$$

Appendix 2(b) Fusing Multiple Local Regions

Next we must fuse together the local region posteriors. We assume that the different local regions are independent, and apply the logarithmic opinion pool once again. We note this assumption is an approximation as local regions are not necessarily independent.

$$P(\Phi, p \mid \Omega) = \prod_{n=1}^{N} P(\Phi, p \mid \Omega_n)$$

where N is the number of local regions. In order to remove possible influence of the number of local regions we divide by the number of local regions. We apply minus the logarithm we get:

$$E = -\frac{1}{N} \log P(\Phi, p \mid \Omega) = -\frac{1}{N} \log \prod_{n=1}^{N} P(\Phi, p \mid \Omega_n) \tag{32}$$

$$= -\frac{1}{N} \log \prod_{n=1}^{N} \prod_{i=1}^{K} \sum_{j=f,b} P(\boldsymbol{x} \mid \Phi, p, M_{n_j}) P(M_{n_j} \mid \boldsymbol{y})$$

$$= -\frac{1}{N}\sum_{n=1}^{N}\sum_{i=1}^{K}\log\left[P_{f_n}H_\epsilon(\Phi(\boldsymbol{x}_i)\boldsymbol{B}_n(\boldsymbol{x}_i)\right.$$
$$\left.+\ P_{b_n}\left(1 - H_\epsilon(\Phi(\boldsymbol{x}_i))\right)\boldsymbol{B}_n(\boldsymbol{x}_i)\right] \quad (33)$$

Equivalently, this may written as:

$$E = \frac{1}{N}\sum_{n=1}^{N}E_n \quad (34)$$

$$E = -\frac{1}{N}\sum_{n=1}^{N}\sum_{\boldsymbol{x}_i\in\Omega}\log\left[P_{f_n}H_\epsilon\left(\Phi(\boldsymbol{x}_i)\right)\right.$$
$$\left.+P_{b_n}\left(1 - H_\epsilon\left(\Phi(\boldsymbol{x}_i)\right)\right)\right]\boldsymbol{B}_n(\boldsymbol{x}_i) \quad (35)$$

References

Arie-Nachimson, M., & Basri, R. (2009). Constructing implicit 3d shape models for pose estimation. In *ICCV* (pp. 1341–1348). IEEE, Piscataway.

Bibby, C., & Reid, I. (2008). Robust real-time visual tracking using pixel-wise operators. *ECCV*, *5303*, 831–844.

Brachmann, E., Krull, A., Michel, F., Gumhold, S., Shotton, J., & Rother, C. (2014). Learning 6D object pose estimation using 3D object coordinates. In *Proceedings, Part II: Computer Vision—ECCV 2014–13th European Conference, Zurich* (pp. 536–551). September 6–12, 2014.

Brox, T., Rosenhahn, B., Gall, J., & Cremers, D. (2010). Combined region and motion-based 3D tracking of rigid and articulated objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *32*(3), 402–415.

Chan, T. F., & Vese, L. A. (2001). Active contours without edges. *IEEE Transactions on Image Processing*, *10*(2), 266–277.

Dambreville, S., Sandhu, R., Yezzi, A., & Tannenbaum, A. (2010). A geometric approach to joint 2D region-based segmentation and 3D pose estimation using a 3D shape prior. *SIAM Journal on Imaging Sciences*, *3*, 110–132.

Dame, A., Prisacariu, V. A., Ren, C. Y., & Reid, I. (2013). Dense reconstruction using 3D object shape priors. In *CVPR* (pp. 1288–1295). IEEE, Piscataway.

Harris, C., & Stennet, C. (1990). RAPiD—a video-rate object tracker. In *British Machine Vision Conference* (pp. 73–77).

Hinterstoisser, S., Lepetit, V., Ilic, S., Holzer, S., Bradski, G., Konolige, K., et al. (2012). Model based training, detection and pose estimation of texture-less 3D objects in heavily cluttered scenes. *Computer Vision ACCV*, *7724*, 548–562.

Horbert, E., Rematas, K., & Leibe, B. (2011). Level-set person segmentation and tracking with multi-region appearance models and top-down shape information. In *IEEE International Conference on Computer Vision (ICCV)* (pp. 1871 – 1878).

Lankton, S., & Tannenbaum, A. (2008). Localizing region-based active contours. *IEEE Transactions on Image Processing*, *17*, 2029–2039.

Leibe, B., Leonardis, A., & Schiele, B. (2004). Combined object categorization and segmentation with an implicit shape model. In *ECCV Workshop on Statistical Learning in Computer Vision* (pp. 17–32).

Lepetit, V., & Fua, P. (2005). Monocular model-based 3D tracking of rigid objects: A survey. *Foundations and Trends in Computer Graphics and Vision*, *1*, 1–89.

Lowe, D. G. (1987). Three-dimensional object recognition from single two-dimensional images. *Artificial Intelligence*, *31*(3), 355–395.

Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, *60*(2), 91–110.

Ma, Y., Soatto, S., Kosecka, J., & Sastry, S. (2003). *An invitation to 3D vision: From images to geometric models*. Heidelberg: Springer.

MathWorks. (2014). *Parallel computing toolbox (R2014a)*. Natick, MA: The MathWorks Inc.

Prisacariu, V., Kahler, O., Murray, D., & Reid, I. (2013). Simultaneous 3D tracking and reconstruction on a mobile phone. In *IEEE International Symposium on Mixed and Augmented Reality (ISMAR)* (pp. 89–98). doi:10.1109/ISMAR.2013.6671768.

Prisacariu, V. A., & Reid, I. D. (2012). Pwp3d: Real-time segmentation and tracking of 3D objects. *International Journal of Computer Vision*, *98*, 335–354.

Rosenhahn, B., Brox, T., & Weickert, J. (2007). Three-dimensional shape knowledge for joint image segmentation and pose tracking. *International Journal of Computer Vision*, *73*(3), 243–262.

Savarese, S., & Li, F. F. (2007). 3D generic object categorization, localization and pose estimation. In *ICCV* (pp. 1–8).

Schmaltz, C., Rosenhahn, B., Brox, T., Cremers, D., Weickert, J., Wietzke, L., et al. (2007). Region-based pose tracking. *IbPRIA (2)* (Vol. 4478, pp. 56–63)., Lecture Notes in Computer Science Berlin: Springer.

Schmaltz, C., Rosenhahn, B., Brox, T., & Weickert, J. (2009). Localised mixture models in region-based tracking. In *Pattern Recognition (Proceedings of DAGM)*. Lecture Notes in Computer Science, Springer, Berlin. Retrieved from http://lmb.informatik.uni-freiburg.de//Publications/2009/Bro09c.

Tan, D. J., & Ilic, S. (2014). Multi-forest tracker: A chameleon in tracking. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.