

Evaluating the Fuzzy Coverage Model for 3D Multi-Camera Network Applications

Aaron Mavrinac, Jose Luis Alarcon Herrera, and Xiang Chen

University of Windsor, Windsor, Ontario, Canada,
{mavrin1,alarconj,xchen}@uwindsor.ca

Abstract. An intuitive three-dimensional task-oriented coverage model for 3D multi-camera networks based on fuzzy sets is presented. The model captures the vagueness inherent in the concept of visual coverage, with a specific target of the feature detection and matching task. The coverage degree predicted by the model is validated against various multi-camera network configurations using the SIFT feature detection and description algorithm.

1 Introduction

Three-dimensional multi-camera networks (that is, camera networks with three-dimensional sensing capabilities) cover a major subset of camera network applications. A fuzzy representation lends itself well to modeling the vague concept of task-specific coverage in such a network. Spatial points with associated direction form the universal set of observable information, and assigning to these a degree of membership in the set of covered points yields a quantitative metric for coverage.

In other work, we shall present a full theoretical derivation of the *fuzzy coverage model*. It is a comprehensive three-dimensional coverage model for cameras and multi-camera networks, based on fuzzy sets, which is derived from well-studied parameters of the standard imaging system model (obtained from internal and external calibration), and can be tuned for specific tasks via a small number of intuitive application parameters. The model takes its inspiration, in part, from work by Cowan and Kovesi [1] and others. Here, we focus on describing and evaluating the 3D multi-camera network coverage model, and dispense with much of the background theory as it is beyond our scope.

A major task requirement for most 3D applications is the detection and matching of local features. Modern popular algorithms such as SIFT [2], SURF [3], MSER [4], and others allow for fairly reliable sparse feature matching between relatively widely separated views – and, therefore, wide-baseline stereo reconstruction. In order to evaluate the coverage of arbitrary scene features, the nature of which (including planar orientations) are not known a priori, directional coverage must be modeled in the scene. As will be shown, the fuzzy coverage model captures this aspect of coverage quite well.

The remainder of this paper is organized thus. Section 2 presents some of the underlying theory and conventions used in our model. Section 3 describes

the model specifically for 3D multi-camera networks, with brief insights into the fundamental computer vision theory. Experimental results for the feature detection and matching task are presented in Section 4. Finally, concluding remarks are given in Section 5.

2 Definitions and Conventions

2.1 Geometric Conventions

When dealing with a three-dimensional Euclidean space, we express positions and orientations in a right-handed Cartesian coordinate system, with the axes of its basis denoted x , y , and z , and fixed-axis rotation angles about these axes denoted θ , ϕ , and ψ , respectively.

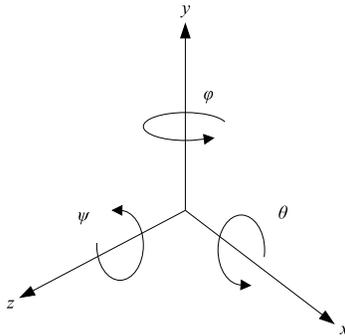


Fig. 1. Axes and Rotations

Naturally, for the spatial coordinates, $(x, y, z) \in \mathbb{R}^3$. Similarly, for the rotation angles, $(\theta, \phi, \psi) \in [0, 2\pi)^3$; for every equivalence class on angles $\bar{\theta}_{2\pi}$ we consider the value $\theta \in [0, 2\pi)$.

For vector direction, we also employ a two-angle orientation defined by inclination angle $\rho \in [0, 2\pi)$, measured from the z -axis zenith, and azimuth angle $\eta \in [0, 2\pi)$, measured from the x -axis in the x - y plane (in the direction of ψ). Since, for $0 \leq e \leq \pi$, $(\pi - e, \eta) \sim (\pi + e, \eta + \pi)$, we generally restrict ρ to $[0, \pi]$ where possible. Note also that $(0, \eta_1) \sim (0, \eta_2)$ and $(\pi, \eta_1) \sim (\pi, \eta_2)$ for any η_1 and η_2 .

Definition 1 *The directional space $\mathbb{D} = \mathbb{R}^3 \times [0, \pi] \times [0, 2\pi)$ consists of three-dimensional Euclidean space plus direction, with elements of the form (x, y, z, ρ, η) .*

For convenience, we denote the spatial component $\mathbf{p}_s = (x, y, z)$ and the directional component $\mathbf{p}_d = (\rho, \eta)$.

A standard 3D pose $P : \mathbb{R}^3 \rightarrow \mathbb{R}^3$, consisting of rotation matrix \mathbf{R} and translation vector \mathbf{T} , may be applied to $\mathbf{p} \in \mathbb{D}$. The spatial component is transformed as usual, i.e. $P(\mathbf{p}_s) = \mathbf{R}\mathbf{p}_s + \mathbf{T}$. The direction component is transformed as follows. If \mathbf{d} is the unit vector in the direction of \mathbf{p}_d , then $P(\mathbf{p}_d) = (\arccos(\mathbf{R}\mathbf{d}_z), \arctan 2(\mathbf{R}\mathbf{d}_y, \mathbf{R}\mathbf{d}_x))$, where $\arctan 2$ is an arctangent variant returning the angle from the x -axis to a vector in the full range $[0, 2\pi)$.

2.2 Fuzzy Sets

A fuzzy set [5] is a pair (S, μ) , where S is a set (called the *universal set*) and $\mu : S \rightarrow [0, 1]$ is a membership function indicating the grade of membership of elements in S to the fuzzy set. For a fuzzy set $A = (S_A, \mu_A)$, the set $\text{supp}(A) = \{x \in S_A | \mu_A(x) > 0\}$ is called the *support* of A , and the set $\text{kern}(A) = \{x \in S_A | \mu_A(x) = 1\}$ is called its *kernel*.

The standard *fuzzy union* operation is defined for fuzzy sets A and B as $A \cup B = (S_A \cup S_B, \mu_{A \cup B})$, where $\mu_{A \cup B}(x) = \max(\mu_A(x), \mu_B(x))$ for $x \in S_A \cup S_B$. Similarly, the standard *fuzzy intersection* operation is defined as $A \cap B = (S_A \cap S_B, \mu_{A \cap B})$, where $\mu_{A \cap B}(x) = \min(\mu_A(x), \mu_B(x))$ for $x \in S_A \cap S_B$. Other t-conorms and t-norms yield different union and intersection operations [6]; in particular, we also use the algebraic product t-norm for intersection, for which $\mu_{A \cap B}(x) = \mu_A(x) \cdot \mu_B(x)$.

In modeling the coverage of cameras and multi-camera networks, we deal with fuzzy subsets of \mathbb{R}^3 and \mathbb{D} .

A continuous fuzzy set A has a discrete counterpart, denoted \dot{A} , which is formed by sampling a finite number of elements from A such that $\mu_{\dot{A}}(\mathbf{p}) = \mu_A(\mathbf{p})$ for all $\mathbf{p} \in S_{\dot{A}}$. We adopt this notation for all discrete fuzzy sets. A discrete fuzzy set \dot{A} has *scalar cardinality* $|\dot{A}| = \sum_{x \in S_{\dot{A}}} \mu_{\dot{A}}(x)$.

3 The Fuzzy Coverage Model

3.1 Single Camera

Consider a fuzzy subset $C \in \mathbb{D}$, with \mathbb{R}^3 in the camera frame basis, where membership grade $\mu_C(\mathbf{p})$ indicates “good imaging” at spatial point \mathbf{p}_s from direction \mathbf{p}_d . In other words, C is the fuzzy set of directional points (in the camera frame) which are *covered*.

Our realization of C is entirely determined from nine intrinsic parameters of the imaging system ($A, f, s_u, s_v, o_u, o_v, w, h$, and z_S) and five application parameters ($\gamma, R_1, R_2, c_{\max}$, and ζ) for a total of 14 parameters. The intrinsic parameters are normally obtained from camera specifications and calibration, and the application parameters are intuitive to set based on task requirements. Four individual factors – visibility, resolution, focus, and direction – yield component fuzzy subsets of \mathbb{D} . These components are combined via algebraic product fuzzy intersection:

$$C = C_V \cap C_R \cap C_F \cap C_D \quad (1)$$

Conveniently, if it is not desirable to account for a component in a certain application, it can simply be left out of (1). In the particular case where directional visibility is not relevant (thus $C = C_V \cap C_R \cap C_F$), C can be treated as a fuzzy subset of \mathbb{R}^3 rather than of \mathbb{D} for practical purposes, since, as will be seen, the directional components have no effect.

Visibility Visibility is a bivalent condition depending whether \mathbf{p} is within the field of view, a function of x , y , and z . However, points near the edges may not be considered “fully visible” for an application’s purposes, so a general-purpose application parameter γ is introduced to fuzzify visibility into a trapezoidal membership function. The value of γ is simply the number of pixels in from the edge of the image where features are considered to be fully within the field of view.

C_V has $\text{kern}(C_V) = \{(x, y, z, \rho, \eta) \in \mathbb{D} \mid -\sin(\alpha_{hl}) + \gamma_h \leq x \leq \sin(\alpha_{hr}) - \gamma_h, -\sin(\alpha_{vl}) + \gamma_v \leq y \leq \sin(\alpha_{vr}) - \gamma_v, z > 0\}$ and $\text{supp}(C_V) = \{(x, y, z, \rho, \eta) \in \mathbb{D} \mid -\sin(\alpha_{hl}) \leq x \leq \sin(\alpha_{hr}), -\sin(\alpha_{vl}) \leq y \leq \sin(\alpha_{vr}), z > 0\}$, where γ_h and γ_v are calculated from γ as follows:

$$\gamma_h = \frac{2\gamma}{w} \sin\left(\frac{\alpha_h}{2}\right) \quad (2)$$

$$\gamma_v = \frac{2\gamma}{h} \sin\left(\frac{\alpha_v}{2}\right) \quad (3)$$

Here, α_h and α_v are the horizontal and vertical angles of view (computed from calibration), with α_{hl} , α_{hr} , α_{vl} , and α_{vr} being the half-angles measured from the principal axis.

Resolution Resolution is inversely proportional to the z -coordinate of \mathbf{p} . C_R has $\text{kern}(C_R) = \{(x, y, z, \rho, \eta) \in \mathbb{D} \mid 0 \leq z \leq z_1\}$ and $\text{supp}(C_R) = \{(x, y, z, \rho, \eta) \in \mathbb{D} \mid 0 \leq z \leq z_2\}$.

The values of z_1 and z_2 are computed from application parameters R_1 and R_2 as follows:

$$z_R = \frac{1}{R} \min \left[\frac{w}{2 \sin(\alpha_h/2)}, \frac{h}{2 \sin(\alpha_v/2)} \right] \quad (4)$$

where w and h are the pixel width and height of the image, respectively. R_1 is the minimum pixel resolution at which the application is not negatively affected by the resolution, and R_2 is the absolute minimum pixel resolution beyond which the application does not produce acceptable results.

Focus Focus is also a function of z . C_F has $\text{kern}(C_F) = \{(x, y, z, \rho, \eta) \in \mathbb{D} \mid z_{\triangleleft} \leq z \leq z_{\triangleright}\}$ and $\text{supp}(C_F) = \{(x, y, z, \rho, \eta) \in \mathbb{D} \mid z_n \leq z \leq z_f\}$.

For $z \in [z_{\triangleleft}, z_{\triangleright}]$, the digital image is effectively in perfect focus; the values are calculated as follows:

$$z_{\triangleleft} = \frac{Afz_S}{Af + s(z_S - f)} \quad z_{\triangleright} = \frac{Afz_S}{Af - s(z_S - f)} \quad (5)$$

where A is the aperture diameter, f is the focal length, z_S is the focus distance, and s is the real size of one pixel.

The values of z_n and z_f are the near and far limits of depth of field for the maximum circle of confusion diameter, application parameter c_{\max} :

$$z_n = \frac{Afz_S}{Af + c_{\max}(z_S - f)} \quad z_f = \frac{Afz_S}{Af - c_{\max}(z_S - f)} \quad (6)$$

Direction Directional visibility is a function of inclination angle ρ . An application parameter ζ is introduced to fuzzify directional visibility into a trapezoidal membership function, which can be used to account for the quality of coverage of an object face dependent on viewing angle, or for the ability to match features.

The fuzzy subset for direction C_D has $\text{kern}(C_D) = \{(x, y, z, \rho, \eta) \in \mathbb{D} \mid \rho \geq \pi/2 + \Theta + \zeta\}$ and $\text{supp}(C_D) = \{(x, y, z, \rho, \eta) \in \mathbb{D} \mid \rho \geq \pi/2 + \Theta\}$, where

$$\Theta = \left(\frac{y}{r} \sin \eta + \frac{x}{r} \cos \eta \right) \arctan \left(\frac{r}{z} \right) \quad (7)$$

with $r = \sqrt{x^2 + y^2}$ and for $z > 0$.

The value of ζ is in the range $[0, \pi/2]$, and reflects the angle of the surface normal of a feature, relative to the principal axis, at which performance for a given task begins to degrade.

3.2 3D Multi-Camera Network

In a 3D multi-camera network, coverage depends on at least two cameras imaging the same point (including direction). Typically, applications perform 3D reconstruction or other pairwise processing of shared scene.

In-Scene Single-Camera Model The *in-scene model* for a single camera, C^s , is simply the camera model C transformed to the world frame. Thus, C^s has six additional parameters – the pose, or extrinsic parameters, of the camera, defined by x , y , z , θ , ϕ , and ψ – for a total of 20 parameters.

Adding the Occlusion Constraint In order to introduce the additional constraint of scene occlusion, the scene model S is required. This consists of a series of opaque plane segments (surfaces of static scene objects). A spatial point \mathbf{p}_s is occluded iff there exists a unique point of intersection $\mathbf{p}_i \neq \mathbf{p}_s$ between the line segment from \mathbf{p}_s to the camera's principal point \mathbf{p}_p and any opaque plane segment.

$$\mathbf{p}_i = \mathbf{p}_s + (\mathbf{p}_p - \mathbf{p}_s)t \quad (8)$$

where t is found, given three non-collinear points P_0 , P_1 , and P_2 in a plane $P \in S$, from

$$\begin{bmatrix} t \\ u \\ v \end{bmatrix} = \begin{bmatrix} \mathbf{p}_{sx} - \mathbf{p}_{px} & P_{1x} - P_{0x} & P_{2x} - P_{0x} \\ \mathbf{p}_{sy} - \mathbf{p}_{py} & P_{1y} - P_{0y} & P_{2y} - P_{0y} \\ \mathbf{p}_{sz} - \mathbf{p}_{pz} & P_{1z} - P_{0z} & P_{2z} - P_{0z} \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{p}_{sx} - P_{0x} \\ \mathbf{p}_{sy} - P_{0y} \\ \mathbf{p}_{sz} - P_{0z} \end{bmatrix} \quad (9)$$

The in-scene model with occlusion is denoted C^o .

Pairwise Coverage Because the 3D multi-camera network performs pairwise processing, we must consider the coverage models of *pairs* of cameras. The pairwise coverage model for cameras k and l is a fuzzy subset $C_{kl}^o = C_k^o \cap C_l^o$.

Network coverage for a 3D multi-camera network can be modeled as a fuzzy subset $C_N = \bigcup_{k,l \in N^2} C_{kl}^o$.

4 Experimental Application

4.1 Feature Matching

Many multi-camera network applications, particularly those involving 3D reconstruction, rely heavily on matching local image features. A variety of feature detection and description algorithms exist, but a major common limitation is degradation of performance over large rotational transformations of the viewpoint [7–10].

Conceptually, the membership degree of a \mathbb{D} vector in C_{kl}^o will depend heavily on the rotation component of P_{kl} : the larger the angle between the principal axes of k and l , the smaller the range of directions covered in common. The ζ application parameter is tuned according to the repeatability of a feature detector over such rotation (experimentally, in our case).

In our experiments, we test the detection and matching performance of the popular SIFT [2] algorithm for local feature detection and description. A total of 26 camera views of a scene of known structure containing 56 features with known 3D positions are used to validate the detection and matching performance predicted by the fuzzy coverage model.

4.2 Apparatus

Software Implementation The fuzzy coverage model and fuzzy vision graph have been implemented in an object-oriented software stack using Python [11]. First, we have developed FuzzPy [12], a generic open-source Python library for fuzzy sets and graphs. Using this functionality, we have developed various classes for the fuzzy sets used in the model. The `Camera` class, initialized with the 14 model parameters, returns the μ -value of any spatial point in camera coordinates using continuous trapezoidal fuzzy sets to implement C . The `MultiCameraSimple` and `MultiCamera3D` classes build their respective discrete fuzzy sets from `Camera` objects and a supplied scene model S . Coverage performance m can be estimated given a discrete fuzzy subset $\hat{D} \in \mathbb{D}$.

Cameras Prosilica EC-1350 cameras, with a sensor resolution of 1360×1024 pixels and square pixel size of $4.65 \mu\text{m}$, are employed. These are fitted with Computar M3Z1228C-MP manual varifocal lenses, with a focal length range of 12 mm to 36 mm and a maximum aperture ratio of 1:2.8. Calibration and image acquisition are performed using HALCON [13]. The SIFT implementation from VLFeat [14] is used for feature detection and matching.

4.3 Experimental Results

An experimentally determined value of $\zeta = 0.3$ is used for the application parameter for directional visibility (other application parameters are $\gamma = 20$, $R_1 = 3.0$, $R_2 = 0.5$, $c_{\max} = 0.0048$).

A four-camera network is used to evaluate the model in this section. One image is taken from each camera, then SIFT is used to find the features in all the images and the matching points in all the pairs for the network. The scene is then simulated and the results are shown in Tables 1 through 6. Figure 2 shows the four images used for this setup, and Figure 3 shows the simulated model used to visualize the predicted performance of this camera setup.

When SIFT is used the result is a bivalent condition that indicates whether the feature, or point in this case, was detected by the algorithm in both images and consequently matched for the given pair. While 1 indicates that the feature was found in both images 0 represents two possibilities: the feature was not detected in any image or it was only detected in one image.

Table 1. Camera Network Pair 1.

Points	P_{06}	P_{13}	P_{14}	P_{15}	P_{38}	P_{39}	P_{46}	P_{47}
SIFT	1	1	1	1	1	1	1	1
Model	0.72	0.00	0.51	0.00	0.70	0.00	0.50	0.58

Table 2. Camera Network Pair 2.

Points	P_{06}	P_{13}	P_{14}	P_{15}	P_{38}	P_{39}	P_{46}	P_{47}
SIFT	1	1	1	1	1	1	1	1
Model	0.94	0.71	0.71	0.24	0.91	0.92	0.82	0.81

Table 3. Camera Network Pair 3.

Points	P_{01}	P_{02}	P_{05}	P_{06}	P_{09}	P_{13}	P_{14}	P_{15}	P_{16}	P_{17}
SIFT	0	0	1	1	0	1	1	1	0	0
Model	0.77	0.81	0.82	0.72	0.28	0.00	0.51	0.00	0.37	0.47

Table 4. Camera Network Pair 3 continued.

Points	P_{18}	P_{26}	P_{27}	P_{28}	P_{37}	P_{38}	P_{39}	P_{46}	P_{47}	P_{48}
SIFT	0	1	1	1	1	1	1	1	1	1
Model	0.46	0.00	0.76	0.68	0.76	0.70	0.00	0.50	0.58	0.51

Table 5. Camera Network Pair 4.

Points	P_{01}	P_{02}	P_{09}	P_{16}	P_{17}	P_{18}	P_{27}	P_{28}	P_{40}	P_{48}
SIFT	0	0	1	0	1	1	1	1	0	1
Model	0.77	0.81	0.82	0.37	0.57	0.65	0.62	0.68	0.63	0.51

Table 6. Camera Network Pair 5.

Points	P_{01}	P_{02}	P_{09}	P_{16}	P_{17}	P_{18}	P_{25}	P_{26}	P_{27}	P_{28}	P_{48}	P_{55}
SIFT	1	1	0	1	0	0	0	0	1	1	1	1
Model	0.85	0.81	0.28	0.42	0.47	0.46	0.54	0.55	0.62	0.70	0.91	0.86

For pair 6 no matches were found, that result is reflected in the model since there are no points with $\mu > 0$.

Tables 1 through 6 show that the points with a SIFT condition of 1 also have a $\mu > .5$ for this camera network, this is considered a good prediction from the camera model, however there are some points that have a SIFT condition of 1 and a $\mu = 0$, regardless these points are considered as false positives because they do not appear as points in the images and SIFT does not look for any specific shape.

Of the total of points that are in the scene 37.5% appear in the model with a $\mu > .5$ and 33.92% appear with a $\mu > .5$ and were also detected by the SIFT implementation.

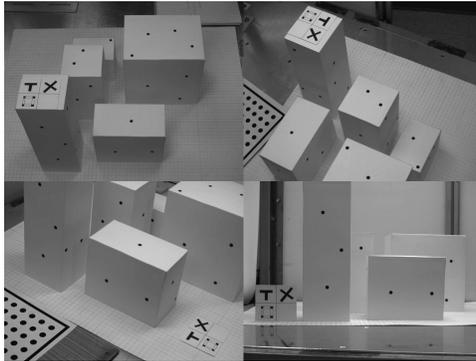


Fig. 2. Views for Camera Network 1

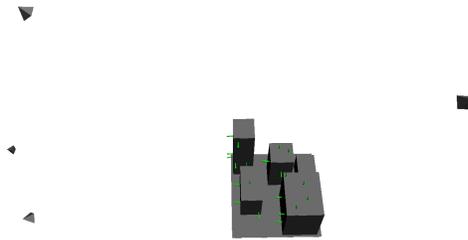


Fig. 3. M^0 for Camera Network 1

Figure 4 shows the matched points for a particular pair of images from a different camera setup. Each camera covers a large amount of features; however, due to the pose of each camera the amount of features covered in common is small. This is reflected in the model shown in Fig 5.

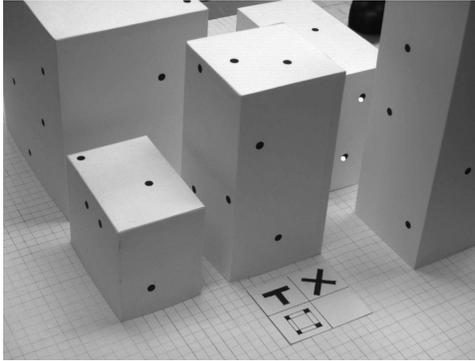


Fig. 4. Matched points for a pair of cameras

In Figure 5 the points being matched by the SIFT implementation are P_{44} and P_{45} ; the μ values for these points are $\mu_{P_{44}} = 0.67$ and $\mu_{P_{45}} = 0.59$.

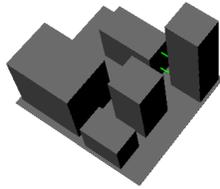


Fig. 5. Model for cameras in Fig. 4

5 Conclusions

The fuzzy coverage model accurately encapsulates information about a 3D multi-camera network's predicted performance on the ubiquitous task of feature detection and matching.

References

1. Cowan, C.K., Kovese, P.D.: Automatic Sensor Placement from Vision Task Requirements. *IEEE Trans. on Pattern Analysis and Machine Intelligence* **10**(3) (1988) 407–416
2. Lowe, D.G.: Distinctive Image Features from Scale-Invariant Keypoints. *Intl. Jnl. of Computer Vision* **60**(2) (2004) 91–110
3. Bay, H., Tuytelaars, T., Van Gool, L.: SURF: Speeded Up Robust Features. In: *Proc. 9th European Conf. on Computer Vision*. (2006) 404–417
4. Matas, J., Chum, O., Urban, M., Pajdla, T.: Robust Wide Baseline Stereo from Maximally Stable Extremal Regions. In: *In Proc. British Machine Vision Conf.* (2002) 384–393
5. Zadeh, L.A.: Fuzzy Sets. *Information and Control* **8**(3) (1965) 338–353
6. Klir, G.J., Yuan, B.: *Fuzzy Sets and Fuzzy Logic: Theory and Applications*. Prentice Hall (1995)
7. Baumberg, A.: Reliable Feature Matching Across Widely Separated Views. In: *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*. (2000) 1774–1781
8. Fraundorfer, F., Bischof, H.: Evaluation of Local Detectors on Non-Planar Scenes. In: *Proc. 28th Wkshp. of the Austrian Association for Pattern Recognition*. (2004) 125–132
9. Mikolajczyk, K., Tuytelaars, T., Schmid, C., Zisserman, A., Matas, J., Schaffalitzky, F., Kadir, T., Van Gool, L.: A Comparison of Affine Region Detectors. *Intl. Jnl. of Computer Vision* **65**(1) (2006) 43–72
10. Moreels, P., Perona, P.: Evaluation of Features Detectors and Descriptors Based on 3D Objects. *Intl. Jnl. of Computer Vision* **73**(3) (2007) 263–284
11. Python Software Foundation: Python
12. Mavrinac, A., Spriet, X.: FuzzPy
13. MVTec Software GmbH: HALCON
14. Vedaldi, A., Fulkerson, B.: VLFeat: An Open and Portable Library of Computer Vision Algorithms (2008)