# Manifold Learning Approach toward Image Feature-based State Space Construction

Yuichi Kobayashi[1], Ryosuke Matsui[2] and Toru Kaneko[1]
[1]*Department of Mechanical Engineering, Graduate School of Engineering, Shizuoka University,*
*3-5-1 Johoku, Naka-ku, Hamamatsu, Japan*
[2]*Nippon Systemware Co. Ltd., 2-15, Nampeidaicho, Shibuya-ku, Tokyo, Japan*

Keywords: Developmental Robotics, Humanoid Robot, Manifold Learning, Image Features.

Abstract: This paper presents a bottom-up approach to building internal representation of an autonomous robot under a stand point that the robot create its state space for planning and generating actions only by itself. For this purpose, image-feature-based state space construction method is proposed using LLE (locally linear embedding). The visual feature is extracted from sample images by SIFT (scale invariant feature transform). SOM (Self Organizing Map) is introduced to find appropriate labels of image features throughout images with different configurations of robot. The vector of visual feature points mapped to low dimensional space express relation between the robot and its environment. The proposed method was evaluated by experiment with a humanoid robot collision classification.

## 1 INTRODUCTION

One of the difficulties which autonomous robots face in non-structured environment is that they are not ready to unexpected factors and changes of their environments. In actual applications, it is not robots themselves but human designers or operators that detect, analyze and find solutions for the unexpected factors. In other words, adaptability of autonomous robots with current technologies is not sufficient as to let them to act in environments close to our daily life. One promising approach to overcome the lack of adaptability of autonomous robots is to build intelligence of robots in a bottom-up manner, known as developmental robotics (Lungarella et al., 2003) and autonomous mental development (Weng et al., 2001). They have common idea for building robot intelligence, e.g., stress on embodiment, self-verification (Stoytchev, 2009), mimicking developmental process of human (infant) (Oudeyer et al., 2007), etc.

Among various concerns in the field of developmental robotics, problem of building state space, with which a robot can plan and control its action, is rather important but has not been gathering sufficient attention. One reason for this is that imitation learning, generating appropriate robot motions based on human demonstration (Argall et al., 2009), is much more effective to generate complex motions with high degrees of freedom. It is known that acquisition of motion without any pre-defined knowledge on robot tasks, e.g., by reinforcement learning (Sutton and Barto, 1998), takes numerous trials and not directly applicable to continuous high-dimensional control problems. The problem of constructing state space, however, is of great importance for autonomous robots to finally generate, control and modify their motions adaptively, even though prototype motion could be built by imitation initially.

Generation of space which is suitable for robot motion learning has been investigated from various viewpoints. Poincaré map is an example of abstract representation for complex robotic behavior learning (Morimoto et al., 2005), where periodic walking pattern by a biped robot was considered. Apart from researches on acquisition of behavior of robot itself, such as walking, jumping, and standing up, state space construction has not been regarded as an important issue. In general, configurations of objects and robots are assumed to be observable in researches on manipulating objects, where positions and postures of objects in the Cartesian (world) coordinate system are used as a solid base.

But in the real world application, measurement of 3D configurations of objects is difficult. It contains difficulties in multiple levels:

1. The framework of 3D configuration measurement

Kobayashi Y., Matsui R. and Kaneko T..
Manifold Learning Approach toward Image Feature-based State Space Construction.
DOI: 10.5220/0004630305290534
In *Proceedings of the 5th International Joint Conference on Computational Intelligence* (NCTA-2013), pages 529-534
ISBN: 978-989-8565-77-8

inherently requires measuring precise shape of an object, but it is difficult to measure whole shape of an object because measurement by camera or laser scanner is normally unilateral.

2. Spatial relation between robot and object is generally very important for both object manipulation and collision avoidance, whereas an object is more likely to be occluded by the robot when the robot is approaching to the object.
3. Deformation of object is normally not considered or requires specific model for mathematical analysis. But it is difficult to precisely model the deformation.

From the viewpoint of the developmental robotics, the 3D representation in the world coordinate is not a sole way to express the state for an autonomous robot. If a robot can build representation of its environment based only on what it can verify by itself, the representation might not suffer from the above-mentioned difficulties (as can be seen in a learning approach (Prankl et al., 2011)).

This paper presents an approach to the interest of *building a representation of a robot* for motion planning and control *without any pre-defined knowledge*. To consider relation between the robot and its environment, image features based on SIFTiScale Invariant Feature Transformj (Lowe, 1999) are used. Image feature-based learning of robot behavior was presented in (Kobayashi et al., 2012), but it did not deal with relation between an object and the robot with a quantitative representation. In this paper, application of a manifold learning is introduced, which enables not only to classify state of the robot but also to evaluate how much the robot is close to a certain state.

Locally Linear Embedding (LLE) (Saul and Roweis, 2003) is used as a means for manifold learning because continuous property of the system can hold only in a local region in the problem of recognition of environment by a robot. For the application of LLE, vector generation based on SIFT-features matching is proposed to deal with a problem that keypoints of SIFT are not consistent through all the images. The proposed framework is evaluated in experiment using a humanoid robot, preceded by preliminary verification of LLE framework with simulated image vectors.

## 2 PROBLEM SETTINGS

Images obtained by CCD camera attached at the head of a robot are considered as input to the robot system, as indicated in Fig. 1. Humanoid robot NAO (Aldebaran Robotics, 2009) is considered both in simulation and experiment. The images contains part of body of the robot, an object which can contact with the robot's body and background which are not affected by configuration of the robot. The configuration of the robot arm is shown in Fig. 2. Shoulder roll joint and shoulder pitch joint are controlled ($\phi_1$ and $\phi_2$), while other two joints are fixed throughout the experiment. This implies that the the motion of the robot arm is constrained on a plane which is vertical to optical axis of the CCD camera. A red plane in Fig. 1 is parallel to the motion constraint plane.

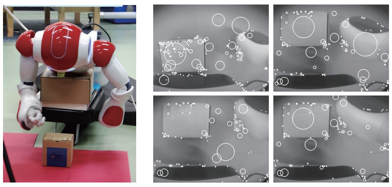

Figure 1: Humanoid robot NAO and its image.

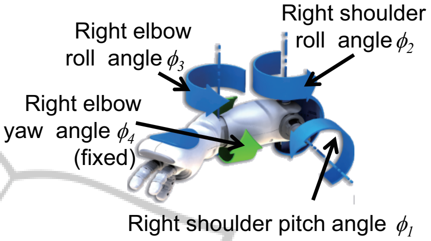
Right elbow roll angle $\phi_3$
Right shoulder roll angle $\phi_2$
Right elbow yaw angle $\phi_4$ (fixed)
Right shoulder pitch angle $\phi_1$

Figure 2: Configuration of robot arm (right).

Image features are extracted from each image, as shown as circles in the right hand of Fig. 1. Keypoints of SIFT (Lowe, 1999) are used as image features. The robot does NOT have knowledge on properties of image features, that is, the robot does not have labels of what is object or what is robot's body in the image in advance. The robot collects images while changing configuration of its arm. Position of the object can also differ irrelevantly to the position of the arm.

Objective of the robot system is to construct a space which provides the following functions:

1. Estimating closeness of its hand to the object
2. Predicting collision of its hand with the object

The first function allows the robot to plan its hand trajectory so as not to be too close to the object, when the robot intends to achieve a task while avoiding collision with obstacles. The second function does not directly allows the robot to avoid collision, but can contribute to the ability by integration of other techniques, e.g, predition of robot's hand in the image space.

# 3 MANIFOLD LEARNING USING IMAGE FEATURES

Manifold learning by LLE is applied to the SIFT keypoints to obtain a continuous space which reflect relation between the hand and its environment. Each keypoint has 128-dimensional feature vector that can be utilized for identification and matching to other keypoints. By the matching process, a keypoint can be traced through multiple images if it is extracted commonly in those images. One problem in generating a vector for manifold learning is that feature vector of a keypoint is not consistent in different images due to change of posture of the arm. The arm, which consists of serial links, inevitably change its posture even when the end of the arm is making translation. Under an assumption that each keypoint tracks a certain part of the arm, a method for matching and labeling is proposed using Self Organizing Map (SOM) (Kohonen, 1995).

Let $D$ denote dimension of image vector, $N$ denote total number of images and $I^{(i)} \in \mathbb{R}^D, i = 1, \cdots, N$ denote vector of image $i$. $M(i)$ denote number of keypoints in image $i$.

## 3.1 Matching and Labeling of Features

First, image vectors $I^{(i)}, i = 1, \cdots, N$ are used to generate a SOM. Let $K$ denotes total number of nodes in SOM. Image vectors are divided into sets by the nodes of the SOM.

$$G(k) = \{i | k = \arg\min_{\ell} \|I^{(i)} - w_\ell\|^2\}, \qquad (3.1)$$

where $w_k \in \mathbb{R}^D$ denotes weight vector of node $k$. $G(k)$ denotes set of images that are similar to $w_k$. For each node, a representative image is decided as

$$\bar{i}_k = \arg\min_{i \in G(k)} \|w_k - I^{(i)}\|^2, \; k = 1, \cdots, K. \qquad (3.2)$$

Image $\bar{i}_k$ is used for generating labels of keypoints. Labels are generated by Algorithm 1.

As a sequel to the labeling procedure, totally $\sum_{k=1}^{K} M(\bar{i}_k)$ labels are generated.

Although feature vector of a keypoint can differ by the change of the robot's configuration, it is likely that feature vectors in images with small differences are similar. By using topological neighbor of SOM, correspondence between keypoint labels can be found. Fig.3 indicates the idea of combining redundant labels. For representative node $\bar{i}_k$ in node $k$, feature vectors of keypoints are averaged within matched keypoints of images $i \in G(k)$. Using the averaged feature vectors, labels are integrated by Algorithm 2.

**Algorithm 1:** Labeling of keypoints.

```
for $k = 1$ to $K$ do
  Select representative image $\bar{i}_k$ for node $k$
  for $\ell = 1$ to $M(\bar{i}_k)$ do
    Select keypoint $\ell$ in image $\bar{i}_k$
    for $i = 1$ to $N$ do
      if $i \notin G(k)$ then
        Apply SIFT matching with keypoint $\ell$ to
        all keypoints in image $i$
        If matching found, label it
      end if
    end for
  end for
end for
```

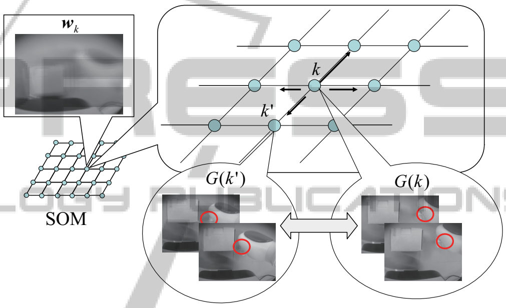

Figure 3: Matching of image features.

By finding correspondence between neighbor nodes, labels which correspond to the same part of the real world are integrated into one label.

## 3.2 Space construction with LLE

Using the obtained labels in the previous section, vectors are defined as follows. Let $L$ denote the number of integrated labels. Keypoint information of image $i$

**Algorithm 2:** Integration of labels.

```
for $k = 1$ to $K$ do
  Find neighbor nodes of node $k$ as $i' \in \mathcal{N}(k)$
  for $\ell = 1$ to $M(\bar{i}_k)$ do
    for $i' = 1$ to $|\mathcal{N}(k)|$ do
      Apply SIFT matching with keypoint $\ell$ by
      average feature vector
      If matching found, record correspondence
      between $\ell$ and the matched label
    end for
    If no matching found, remove label $\ell$
  end for
end for
Integrate all labels using recorded correspondence
```

is converted to vector $x_i \in \mathbb{R}^{2L}$, where $x_i$ is defined by

$$x_i = [u_1^{(i)}\ v_1^{(i)}\ u_2^{(i)}\ v_2^{(i)}\ \cdots\ u_L^{(i)}\ v_L^{(i)}]^T. \tag{3.3}$$

$(u_\ell^{(i)}, v_\ell^{(i)})$ denotes position (image coordinate) of keypoint whose label is $\ell$ in image $i$. If keypoint whose label is $\ell$ does not exist in image $i$, averages over all images are used for $(u_\ell^{(i)}, v_\ell^{(i)})$. Finally, data matrix for LLE is constructed as $H = [x_1\ x_2\ \cdots\ x_N] \in \mathbb{R}^{2L\times N}$. LLE is a method which maps a high-dimensional vector ($2L$ in this application) to a low-dimensional vector while preserving local linear structure of each data around its neighborhood. Weighting coefficient $v_j^i, j = 1, \cdots, n$ for sample $i$, where $n$ denotes the number of neighborhood, is calculated so that the cost function defined by the following is minimized.

$$\varepsilon_1 = \sum_{i=1}^{N} \|x_i - \sum_{j=1}^{n} v_j^i x_j^i\|^2, \tag{3.4}$$

where $x_j^i$ denotes neighborhood sample of $x_i$. A low-dimensional vector $y_i \in \mathbb{R}^d$, corresponding to $x_i$, is calculated so that the following cost function is minimized.

$$\varepsilon_2 = \sum_{i=1}^{N} \parallel y_i - \sum_{j=1}^{n} v_j^i y_j^i \parallel^2, \tag{3.5}$$

where $y_j^i, j = 1, \cdots, d$ denotes neighborhood of $y_i$ and $d$ denotes dimension of the low-dimensional space.

# 4 EXPERIMENT

The proposed representation was evaluated by experiment in two ways, with simulated images and actual images obtained by a CCD camera attached at the head of the robot.

## 4.1 Evaluation with Simulated Image

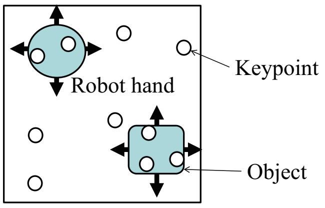

Figure 4: Simulated keypoints.

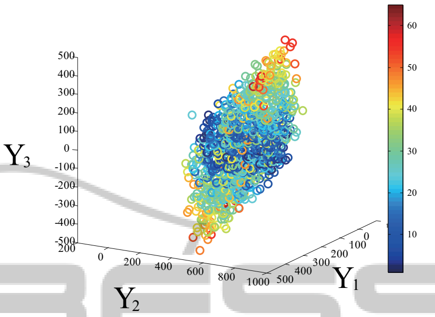

Figure 5: Distance from object.

Fundamental property of LLE was tested in conditions similar to the problem setting. Virtual keypoints are generated as indicated in Fig.4. It was assumed that an object and the robot hand is captured in a image frame of 400×400 [pix] size. There were 10 keypoints to be detected on the object, 10 on the robot hand and 5 in background. The positions of the object and the hand were varied with uniform distribution for collecting samples. Total number of images was set as $N = 1000$. Number of keypoints was set as $m = 25$. Thus, data matrix for LLE was $H \in \mathbb{R}^{50\times 1000}$. Dimension of the mapping was set as $d = 3$. To simulate matching error of keypoints, position information of 10 % of the keypoints in the data vector were removed. That is, 10 % of the elements of $H$ was replaced by the average value of positions of the corresponding keypoint.

The result of mapping by LLE is depicted in Fig.5 and Fig.6. The two figures show the same point information from different perspectives. The colors of the points denote distances between the object and the hand in the corresponding images. It can be seen in the figure that one direction in the feature space reflect the distance between the object and the hand.

## 4.2 Evaluation with Real Image of Humanoid

The evaluation in the previous section did not include keypoints extraction and matching. In the experiment with the humanoid robot, the proposed method described in section was tested. Image size was

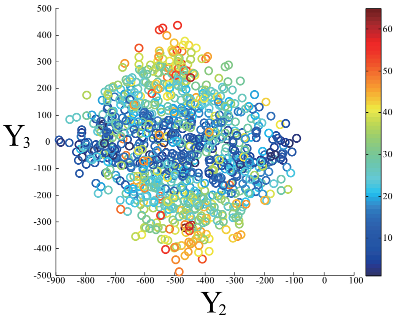

Figure 6: Distance from object.

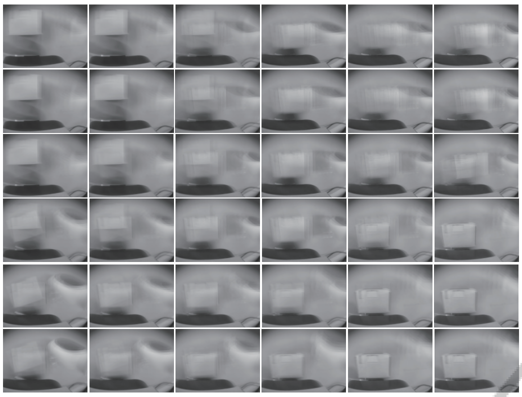

Figure 7: Images corresponding to SOM.

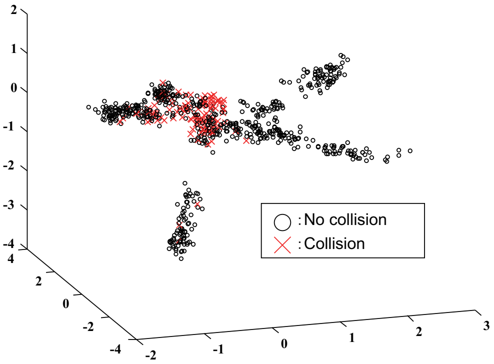

Figure 8: Obtained mapping with LLE.

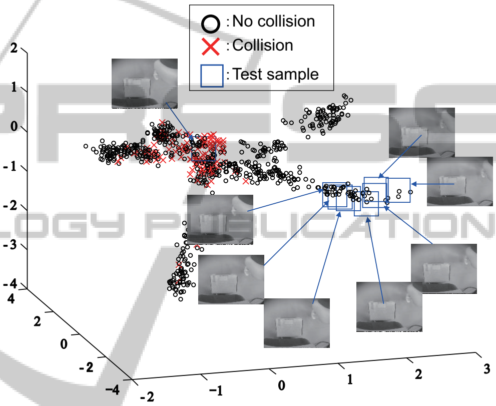

Figure 9: Prediction of collision with test sample.

640×480 The number of nodes of SOM was set as 6×6. Joint angles $\phi_1$ and $\phi_2$ were changed an interval of 2 [deg]. Position of the object was changed simultaneously and totally 732 images were taken.

Fig.7 indicates weight vector of all of the nodes of SOM. It can be seen similar images are located in the neighborhood in the topology of the map. After labeling (Algorithm 1) and integration of labels (Algorithm 2), 1674 labels were obtained.

Fig.8 shows 3-dimensional mapping obtained by the proposed method. Each point (circle or cross) indicates a vector obtained by converting vector $x_i$ by LLE. Cross indicates that the image corresponds to a situation where the hand is contacting with the object. Circle indicates that there is no contact. It can be seen that crosses are concentrating around a certain region. Distance between the object and the hand, however, could not be seen in the obtained map.

Test images, which were not contained in the images for training (generating LLE mapping), were mapped onto the obtained space. Boxes in Fig.9 indicates test samples, where corresponding images are also displayed. It can be seen that image in which the hand is the most distant from the object is located at the furthest from the region with dense crosses. Images in which the hand is closer are gradually located closer in the mapped space. But there is a jump at the last step to contact with the object into the region with dense crosses.

Using the obtained map, classification of collision was evaluated. Collision of the hand with the object was classified by whether a sample is included in the sphere whose center is the average of the crosses. The optimal radius was set as $r = 0.74$. Table 1 shows the classification result.

For comparison, a linear mapping was also implemented. Fig.10 shows the result of mapping with PCA (Principal Component Analysis) using the same data matrix. Crosses, corresponding to contact of the hand with the object, are more dispersing compared with Fig.8. Classification result with PCA is shown in Table 2. It can be seen that consideration of nonlinearity brings conspicuous difference of classification performance.

## 5 DISCUSSION

The labels of keypoints obtained by the proposed method was still numerous even after integration of Algorithm 2. It is possible to consider reliability of the keypoints by evaluating frequency of appearance. It should be also considered that there are not so many

Table 1: Discrimination of collision wit LLE.

| | Collision [%] | No collision [%] |
|---|---|---|
| Recognized as collision | 95 / 115 [82.6] | 111 / 617 [18.0] |
| Recognized as no collision | 20 / 115 [17.4] | 506 / 617 [82.0] |

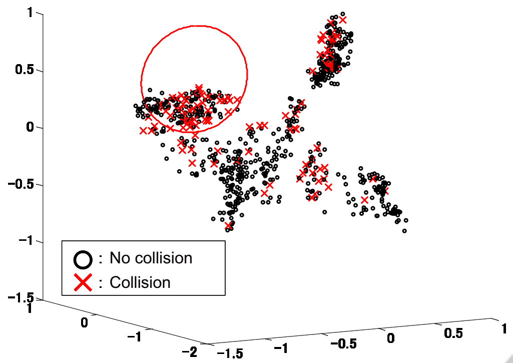

Figure 10: Mapping result with PCA.

Table 2: Discrimination of collision with PCA.

| | Collision [%] | No collision [%] |
|---|---|---|
| Recognized as collision | 63 / 115 [54.8] | 132 / 617 [21.4] |
| Recognized as no collision | 52 / 115 [45.2] | 485 / 617 [78.6] |

keypoints stably detected on the hand of NAO. Not only improving reliability of image features (e.g., using PCA-SIFT (Ke and Sukthankar, 2004)) but also applying multiple kinds of features will be important to generate good data matrix.

Mismatching of keypoints is substantially inevitable when a part of the robot changes its posture. Therefore, it will be important to expand the framework to a more flexible one, which can continuously map a vector whose elements are partly lost.

## 6 CONCLUSIONS

In this paper, a manifold learning method was tested for bottom-up acquisition of a space which is useful for motion generation of a robot. This approach does not require any specific knowledge on the robot and its environment, which will contribute to development of truly flexible intelligence of autonomous robots.

In the evaluation of simulated image vectors, it was verified that the distance between the robot hand and the object was reflected in the map. In the evaluation of experiment with real images, the robot could classify images whether the robot is colliding with the object based on the obtained mapping. Moreover, manifold learning turned out to be superior to linear dimensionality reduction, PCA. As a next step, it will be required to extend the idea of bottom-up construction of a low-dimensional space to the case where features frequently disappears.

## ACKNOWLEDGEMENTS

This work was partly supported by Kayamori Foundation of Informational Science Advancement.

## REFERENCES

Aldebaran Robotics (2009). Nao. http://www.aldebaran-robotics.com/. Technical Specifications Document.

Argall, B. D., Chernova, S., Veloso, M., and Browning, B. (2009). A survey of robot learning from demonstration. *Robot. Auton. Syst.*, 57(5):469–483.

Ke, Y. and Sukthankar, R. (2004). Pca-sift: A more distinctive representation for local image descriptors. *Computer Vision and Pattern Recognition*.

Kobayashi, Y., Okamoto, T., and Onishi, M. (2012). Generation of obstacle avoidance based on image features and embodiment. *International Journal of Robotics and Automation*, 24(4):364–376.

Kohonen, T. (1995). *Self-Organizing Maps*. Springer Press.

Lowe, D. G. (1999). Object recognition from local scale-invariant features. In *Proc. of IEEE International Conference on Computer Vision*, volume 2, pages 1150–1157.

Lungarella, M., Metta, G., Pfeifer, R., and Sandini, G. (2003). Developmental robotics: a survey. *Connection Science*, 15:151–190.

Morimoto, J., Nakanishi, J., Endo, G., Cheng, G., Atkeson, C. G., and Zeglin, G. (2005). Poincaré-Map-Based Reinforcement Learning For Biped Walking. In *Proc. of IEEE International Conference on Robotics and Automation*.

Oudeyer, P. Y., Kaplan, F., and Hafner, V. (2007). Intrinsic motivation systems for autonomous mental development. *IEEE Transactions on Evolutionary Computation*, 11(2):265–286.

Prankl, J., Zillich, M., and Vincze, M. (2011). 3d piecewise planar object model for robotics manipulation. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pages 1784 –1790.

Saul, L. K. and Roweis, S. T. (2003). Think globally,fit locally : Unsupervised learning of low dimensional manifolds. *Journal of Machine Learning Research*, 4:119–155.

Stoytchev, A. (2009). Some basic principles of developmental robotics. *IEEE Transactions on Autonomous Mental Development*, 1(2):122–130.

Sutton, R. S. and Barto, A. G. (1998). *Reinforcement Learning: An Introduction (Adaptive Computation and Machine Learning)*. A Bradford Book.

Weng, J., McClelland, J., Pentland, A., Sporns, O., Stockman, I., Sur, M., and Thelen, E. (2001). Autonomous mental development by robots and animals. *Science*, 291:599–600.