

A Component-based Architecture for Vision-based Gesture Recognition

Farhad Dadgostar, Abdolhossein Sarrafzadeh

Institute of Information and Mathematical Sciences, Massey University
Auckland, New Zealand

Email: F.Dadgostar@massey.ac.nz, H.A.Sarrafzadeh@massey.ac.nz

Abstract

In this paper we present our research on developing a vision-based gesture recognition system. This system has three abstract layers each with their own specific type and requirements of data which enable us to implement them as separate components. The first layer is the skin detection layer. One of our contributions to the research is an adaptive skin detection algorithm that forms the core of this layer. This component provides a set of disperse skin pixels for a tracker that forms the second layer. This second component is based on the Mean-shift algorithm which has been improved for robustness against noise using our novel fuzzy-based edge estimation method that makes the tracker suitable for real world applications. The third component is the gesture recognition layer which is based on a gesture modelling technique and a classification method that we have developed for this purpose. We have used the angle space to model the input gesture and artificial neural-networks for classification of the gesture.

Keywords: Adaptive skin detection, hue colour space, mean-shift algorithm, object tracking, gesture recognition, neural-networks

1 Introduction

Gesture recognition and gesture-based interaction is becoming an increasingly attractive research subject in human-computer interaction (HCI). There is a wide range of applications for this technology, including interactive games, performance analysis, surveillance monitoring, disability support, virtual reality and many others.

Although there are some commercially available solutions based on Digital Gloves and Body Markers, at the time this research is being undertaken, they are very expensive and also require physical attachment to the body. Consequently, these technologies are less available for general use and are sometimes seem as intrusive from the user's point of view.

Vision-based gesture recognition is an alternative approach which holds potential for being made available for a wider range of uses. Furthermore, it can also be applied in applications like surveillance monitoring, or video analysis in which the human subject is not accessible.

Vision-based gesture recognition has been a challenging research. It requires the movement information of various body parts like the hands and head which is not easily extractable from a 2D image. On the other hand for most of the applications, being real-time is an important feature which makes the approaches requiring time-consuming computations

less favourable. In addition, acceptability of a new technology in real world applications depends on its adaptability to the current technology and the cost.

Based on the discussed features for a gesture recognition system, we took a component-based approach to system development. The system, contains three major components: i) The skin detection, ii) hand and face tracking, and iii) gesture recognition. In the next section, we describe the research background forming the foundations of each component. In section 3, we discuss the components we have developed and the type of information which is transmitted between them. Discussion and conclusion are presented in Section 4.

2 Research Background

The components which are required for a vision-based gesture recognition system basically need different types of information. For instance object detection and tracking is a subject that falls more in the area of computer vision, but gesture recognition is a more abstract subject which may have a wider range of applications.

Obviously the first step in a vision-based system is detection and tracking the desired object. The latest trend in object recognition is the use of AI techniques to train classifiers. For some time the focus of the work was on the training algorithms themselves. However, the influence of the choice of features and

the quality of the training image set cannot be underestimated. For feature selection, there are two main approaches: i) invariant features and ii) non-invariant features.

Features which are dependent on the geometry or the location of the object are called non-invariant. Edges and corners can be considered in this feature category. They do have the weakness of demanding the tuning of specific parameters. A more robust set of features that somehow assesses edges and corners and relates their presence to certain regions of the image would be useful. Such features exist and are called Haar-like features. Viola and Jones [1] were the first people who developed a robust real-time algorithm based on Haar-like features. Non-invariant features are not robust against modifiers like change of viewpoint or rotation, which specifically makes them unsuitable for tracking an articulate object like hand.

The invariant features are those features which are not affected by certain changes in the object viewpoint. It is widely accepted that invariant features would be independent from modifiers such as translation, scaling, rotation and lighting conditions [2]. Rather, there are features that are more or less robust to one or more modifiers. One of the simplest sets of invariant features are histograms. Although, for generic computer vision applications, histograms are not necessarily good features, they are quite useful under the right circumstances.

2.1 Skin colour segmentation

Skin colour segmentation has shown promising results for hand/face detection and tracking with different colour spaces including RGB, ICrCb, HSV, HSI, HS and IUV. The main idea in these approaches is using a set of manually segmented skin images, to find a region in the colour space and using the result as a colour skin probability density function. This procedure is called "training". The result of training in most colour spaces is a single connected region in which boundaries can be specified by a limited number of surfaces, lines or points. RGB, HS and Hue colour spaces have shown the best results for colour skin segmentation.

Bradski [3] has used Hue colour space to find the centre of the mass of the skin pixels for detection of the computer user's face. Kolsch and Turk [4] have used a similar approach for detecting a group of features they called "flocks of features", for hand tracking. Ruiz-del-Solar and Verschae [5] have used this technique together with a fuzzy approach for calculating the membership degree of a pixel to the colour skin set based on its probability density and its neighbours' probability density. Imagawa, Lu and Igi [6] have used a mixed approach based on locating the face, using non-invariant features and estimating the colour probability density function for segmenting hands.

2.2 Tracking using the Mean-shift Algorithm

The result of the skin detection algorithm is a disperse set of pixels on the image which may not provide a rigid space. Therefore the tracking algorithm should be able to track a mass of pixels over time. One of the most computationally efficient algorithms for this purpose is the Mean-shift algorithms. The Mean-shift algorithm and its application in pattern recognition was originally introduced by Fukunaga and Hostetler [7] in 1975. The general approach for object tracking using the Mean-shift algorithm has been described in Bradski [3]. Some of the applications of the Mean-shift algorithm as a general tool for analyzing the feature space are introduced in Comaniciu [8]. Image segmentation is one of the applications which has attracted researchers, following some successful applications including Cheng's work [9]. Cheng's idea was finding and segmenting the peaks in the histogram of the image and using this information for segmenting the image. Another successful application of the Mean-shift algorithm is object tracking with the core idea of representing the object as a set of features that may vary in number and distance over time [10]. This assumption is realistic in the real-world applications of video analysis, because a 100% accuracy in detecting features is not essential. In 1998, Bradski [3] introduced a new variation of the Mean-shift algorithm for blob tracking in video sequences, called "Continuously Adaptive Mean-shift" or the CAM-Shift algorithm, which is one of the earliest works in application of the Mean-shift algorithm in object tracking. Comaniciu [11] used the Mean-shift algorithm for object tracking with a moving camera. This meant that the feature extraction of the object is more difficult because of the changing of the background, but the basic idea of the tracking is the same as Bradski's [3]. Allen [12] used the CAM-Shift algorithm for tracking of multiple colour patches. Wang, Chen and Huang [13] used the features extracted from the wavelet to track the object.

2.3 Gesture Recognition

The gesture recognition problem consists of pattern representation and recognition. Several methods have been used for gesture recognition: template matching, dictionary lookup, statistical matching, linguistic matching, neural-networks, Hidden-Markov models, and ad hoc methods [14].

Hidden Markov Model (HMM) is used widely in speech recognition, and recently many researchers are applying HMM to temporal gesture recognition. However, because of the difficulty of data collection for training an HMM for temporal gesture recognition, the vocabularies are very limited, and to reach to an acceptable accuracy, a great amount of data is required and a lot of time is spent to estimate the parameters of the HMM. Some researchers have

suggested to use a better approach for more complex systems [15]. However, this remains an open question.

Yang and Xu [14] proposed a method for developing a gesture-based interaction system using a multi-dimensional HMM. They have used the Fast Fourier Transform (FFT) to convert the input gesture to a sequence of symbols to train the HMM. Zhu, Ren, Xu and Lin [16] used visual spatio-temporal features of the gesture for developing a real-time gesture controller which includes visual modelling, analysis, and recognition of continuous dynamic hand gestures.

Watnabe and Yachida [17] proposed a method of gesture recognition from image sequences. The input image is segmented using maskable templates and then the gesture space is constituted by Karhunen-Loeve (KL) expansion using the segment. Su [18] proposed a fuzzy rule-based approach for spatio-temporal hand gesture recognition for sign language detection.

Oka, Satio and Kioke [19] proposed a gesture recognition based on measured finger trajectories for an augmented desk interface system. They used Kalman-Filter for predicting the location of the multiple fingertips and HMM for gesture detection. New, Hasanbelliu and Aguilar [20] proposed a gesture recognition system, based on hand-shape template matching, for hand tracking and detecting the number of fingers being held up, to control an external device. Perrin, Cassinelli, and Ishikawa [15] described a finger tracking gesture recognition system based on a laser tracking mechanism which can be used in hand-held devices. Lementec and Bajcsy [21] proposed an arm gesture recognition algorithm from Euler angles acquired from multiple orientation sensors, for controlling unmanned aerial vehicles in presence of manned aircrew.

3 Vision-based Gesture Recognition

In this section, we introduce the components of our vision-based gesture recognition system. Each component provides the output for the next layer of the application and also can be used individually.

3.1 Skin Detection

The purpose of the first component of the system is a reliable skin detection. The core of this component is our adaptive skin detection algorithm. In ideal conditions (e.g. using a blue background) a non-adaptive skin detection may be sufficient. The adaptive skin detection algorithm was originally introduced by the authors [22, 23]. This is a histogram based segmentation algorithm which adapts its thresholds on the local information extracted from the video sequence. This algorithm has four main steps, as follows:

1) Training the Global Skin Detector: By using a set of training data, the thresholds of the skin colour in hue colour space is recognized. These two thresholds called the Global Skin Detector can detect skin properly, but may falsely detect some non-skin pixels. To reduce the false detection, we introduced the next steps to make this method, adaptive and more accurate.

2) Detection of in-motion skin pixels: The next step is detecting the in-motion pixels of the image and filtering the detected pixels using the Global Skin Detector. For this purpose any kind of motion detection algorithm can be used. However, frame subtraction which is one of the simplest methods, shows acceptable result in case of having fixed camera and static background. Outputs of this step are those pixels which with a higher probability belong to the skin regions of the image.

3) Recalculating the thresholds: In the third step, the pixels that were considered as moving pixels belonging to the user's skin are used for retraining the detector. In this research we have used a histogram of Hue factor as the base for calculating low (T_L) and high (T_U) thresholds for filtering the image. From the in-motion skin pixels, another histogram is extracted, and the second histogram is merged with the original histogram using the following equation:

$$H_{n+1} = (I-A)*H_n + A*H_M$$

- H_{n+1} is the new histogram for skin detection (for the next frame)
- H_n is the histogram for skin detection in the current frame
- H_M is the histogram of the in-motion pixels of the skin colour
- And A , is the weight for merging two histograms.

Empirical results show that a value between 0.02 - 0.05 gives the best output for the final skin detector.

4) Filtering using adaptive skin detector: For each frame, thresholds of the Hue factor are recalculated such that they cover 90% of the area of the new histogram. Finally, the filter for each frame is described as follows.

$$f(I) = \begin{cases} true & \text{if } T_L(H_n) \leq I \leq T_U(H_n) \\ false & \text{else} \end{cases}$$

- I , is the Hue factor for each pixel
- H_n is the Hue histogram for the skin colour
- T_L is the calculated lower threshold for histogram H_n
- T_U is the calculated upper threshold for the histogram H_n .



(a)



(b)

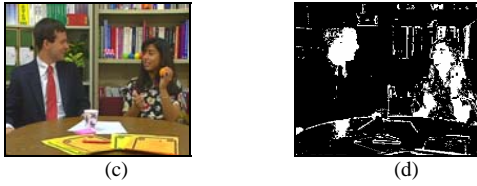


Figure 1. The output of the adaptive skin detection algorithm on the input image sequence: a) first frame, b) first filtered images, c) the last frame (frame 90), d) the last filtered frame

3.2 Object tracking and boundary estimation

After detecting the skin in the image, the next step is tracking the blobs, or flocks of pixels. One of the drawbacks of the fast skin segmentation algorithms, is producing disperse multiple detections. This feature makes the output more favourable for mass tracking algorithms and specifically the Mean-shift algorithm. One of the limitations of the Mean-shift tracking algorithms is the lack estimating the size of the tracked object. The original implementation of the Mean-shift algorithm for object tracking [3], uses a measurement function based on the density of the kernel which is not robust in some conditions.

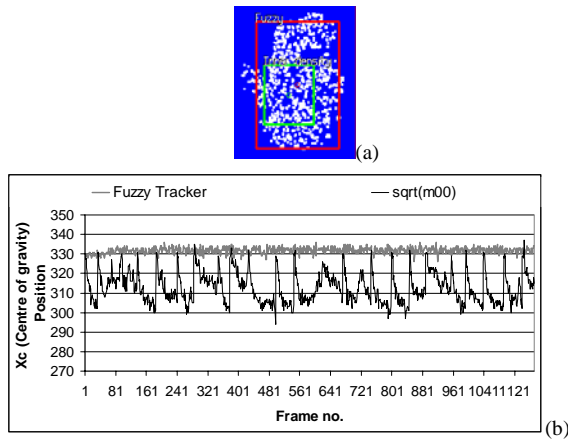


Figure 2. Behaviour of the algorithms with a noise level of 20%, a) The correct detection determined by Edge density-fuzzy, b) smaller rectangle is the result of kernel density-based – $\sqrt{m00}$ – method

Our contribution to the research is the introduction of the fuzzy-based boundary detection for estimating the boundaries of the kernel. The details of the algorithm and the results of its comparison to other algorithms may be found in Dadgostar, Sarrafzadeh and Overmyer [24]. The result of one of the experiments demonstrating the stability of this method against noise is presented in Figure 2. The summary of the fuzzy boundary estimation for the Mean-shift algorithm is as follows:

```

Algorithm FuzzySkinBlobTracker()
Begin
    Kernel = MaximumSize();

    For each frame Imagei
        For each pixel Px,y in Imagei
            If T1 ≤ Hue(Px,y) ≤ T2
                Feature[x, y] = 1;
            Else
                Feature[x, y] = 0;
        End For

        For each boundary Bi of the Kernel
            Delta = FuzzyBoundaryDecision(Bi);
            ChangeBoundarySize( Kernel, Bi, Delta );
        End For

        Repeat
            [M00, M01, M10] = ComputeMoments( Kernel,
            Features );

            Cw = Centre_of( Kernel );
            Cg = (M01/M00, M10/M00);
            ShiftWindow( Kernel, Cg - Cw );
        Until |Cg - Cw| ≤ ε;
    End For
End

```

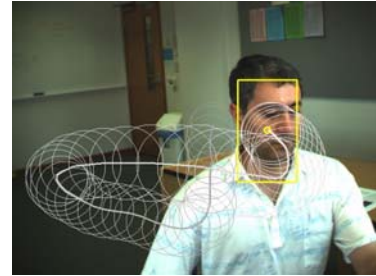


Figure 3. Real-time head tracking, as one of the required inputs for the gesture recognition system

3.3 Gesture Modelling and Interpretation

The result of the tracker is a sequence of coordinates showing the position of the object of interest, over time. The task of the gesture interpretation system is to recognize a gesture pattern in the motion history which is the task of the third component. We have developed a prototype of this component for detecting 3 simple gestures which shows promising results. Development of this component is still an ongoing research.

We have used the angle space for modelling 2D gestures. The angle of the centre of gravity of the tracked object was sampled over time, and quantized to 0 to 35. Therefore an input gesture can be described as a finite set of integer values from 0 to 35 which implicitly includes the time and the direction of the gesture movements. Figure 4a shows a simple hand movement. The density of the arrows in different parts of the movement represents the slower speed of the hand in those parts. It is observable that the hand has had some vibrations in some parts, and in Figure 4a, the number of samples (arrows) is considerably more than those in Figure 4b, which is the quantized version of the original movement. Using this technique, a gesture can be translated into a gesture signal (Figure 4c), which reduces the gesture recognition problem to a signal matching problem.

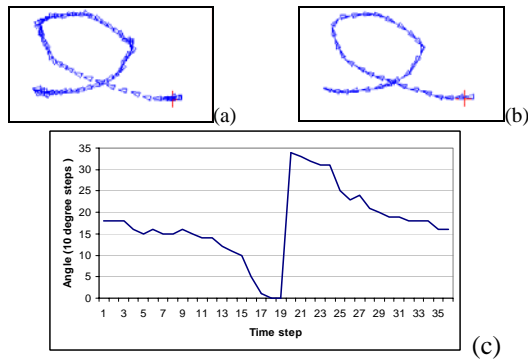


Figure 4. Gesture pattern a) Original gesture move, b) Sampling distance, c) Collected data over time

Gesture recognition is done using a neural-network (NN) with 28 inputs, 3 outputs and 3 inner layers. The outputs are translated to 0/1 as true/false values, using a HardLimit transform (Figure 5). We have trained and tested this NN for detection of 3 different gestures (Table 1).

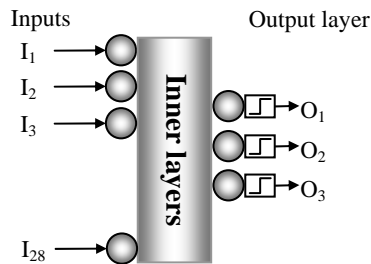


Figure 5. The structure of the NN for gesture classification

Table 1. Gesture detection using NN

Gesture	(1)	(2)	(3)
Size of training data	235	221	254
Average size	31	29	23
Size of the normal vector	28	28	28
Size of the test data	126	113	96
Accuracy of the classifier for each gesture	100%	98.23%	97.917%

Although our original design was based on gesture detection with known boundaries, we also applied the NN with continuous input to observe its validity for gesture detection. In the experiment, the input signals were used as the inputs of the NN. The inputs of the NN were initially set to 0, and each input angle was pushed to the input queue at the time of observation. Figure 6 shows the input gesture and the detection results. The sequence of the three gestures was correctly detected. However, the boundaries of the gesture signal were not determinable because of the multiple detections of each gesture.

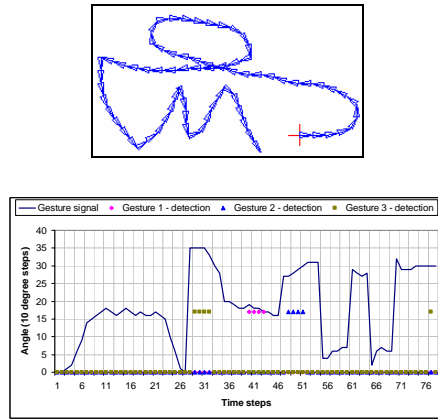


Figure 6. Continues gesture detection using NN

4 Discussion and Conclusion

In this paper, we described the components which we have designed and developed for our vision-based gesture recognition system. Based on this design there are three abstract layers: i) adaptive skin detection and segmentation, ii) hand and face tracking which is done by a fuzzy-based Mean-shift algorithm, and iii) gesture recognition. The adaptive skin detection is based on the hue histogram of the image which adapts itself to the skin colour of the subject in the image sequence. This algorithm is flexible in terms of what the motion detection method it uses. However, the frame subtraction method which is probably one of the simplest methods shows acceptable results.

The second component is the tracking algorithm that is an improved version of the Mean-shift algorithm. This algorithm requires low computation power and can be applied in real-time applications.

The third component is the gesture recognition system which is based on our modelling approach for 2D gestures. We have used a NN for gesture classification. However, the gesture signal can be applied with other classifiers such as support vector machines or Eigen vector-based classifiers.

These components together make our prototype of the gesture recognition system which will itself be used as a component of the Next Generation Intelligent Tutoring System being developed at Massey University in New Zealand with the aim of enabling computer tutors to respond to nonverbal communication.. The gesture recognition system has several applications for this project. Initially it provides a command source for the system such that the user can communicate with it with simple gestures. This facility can particularly be helpful for disabled users and a step toward finding a replacement for mouse and keyboard. The next intended application is using it as one of the sources of information for recognizing the emotional state of the student, through gestures and body language which requires more research but based on the findings in behavioural science seems to be feasible.

5 References

- [1] P. Viola and M. J. Jones, "Robust Real-time Object Detection," Cambridge Research Laboratory February 2001 2001.
- [2] J. Wood, "Invariant pattern recognition: a review," *Pattern Recognition*, vol. 29, pp. 1-17, 1996.
- [3] G. R. Bradski, "Computer Vision Face Tracking For Use in a Perceptual User Interface," *Intel Technology Journal*, 1998.
- [4] M. Kolsch and M. Turk, "Fast 2D Hand Tracking with Flocks of Features and Multi-Cue Integration," presented at Conference on Computer Vision and Pattern Recognition Workshop (CVPRW'04), Washington, D.C., USA, 2004.
- [5] J. Ruiz-del-Solar and R. Verschae, "Skin detection using neighbourhood information," presented at Proceedings of the Sixth IEEE International Conference on Automatic Face and Gesture Recognition, 2004.
- [6] K. Imagawa, S. Lu, and S. Igi, "Color-Based Hands Tracking System for Sign Language Recognition," in *Proceedings of the 3rd. International Conference on Face and Gesture Recognition*: IEEE Computer Society, 1998, pp. p. 462.
- [7] K. Fukunaga and L. Hostetler, "The estimation of the gradient of a density function, with applications in pattern recognition," *IEEE Transactions on Information Theory*, vol. 21, pp. 32-40, 1975.
- [8] D. Comaniciu and P. Meer, "Mean Shift Analysis and Applications," presented at Proceedings of the Seventh IEEE International Conference on Computer Vision, Kerkyra, Greece, 1999.
- [9] Y. Cheng, "Mean shift, mode seeking, and clustering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, pp. 790-799, 1995.
- [10] D. Comaniciu and P. Meer, "Mean Shift: A Robust Approach Toward Feature Space Analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, pp. 603-619, 2002.
- [11] D. Comaniciu, V. Ramesh, and P. Meer, "Real-Time Tracking of Non-Rigid Objects using Mean Shift," presented at IEEE Computer Vision and Pattern Recognition, 2000.
- [12] J. G. Allen, R. Y. D. Xu, and J. S. Jin, "Object Tracking Using CamShift Algorithm and Multiple Quantized Feature Spaces," presented at Workshop on Visual Information Processing, Conferences in Research and Practice in Information Technology, Sydney, Australia, 2003.
- [13] R. Wang, Y. Chen, and T. S. Huang, "Basis Pursuit for Tracking," presented at IEEE International Conference on Image Processing (ICIP'01), Thessaloniki, Greece, 2001.
- [14] J. Yang and Y. Xu, "Gesture Interface: Modeling and Learning," presented at IEEE International Conference on Robotics and Automation, 1994.
- [15] S. Perrin, A. Cassinelli, and M. Ishikawa, "Gesture recognition using laser-based tracking system," presented at Automatic Face and Gesture Recognition, 2004. Proceedings. Sixth IEEE International Conference on, 2004.
- [16] Y. Zhu, H. Ren, G. Xu, and X. Lin, "Toward real-time human-computer interaction with continuous dynamic hand gestures," presented at Proceedings. Fourth IEEE International Conference on Automatic Face and Gesture Recognition, 2000.
- [17] T. Watanabe and M. Yachida, "Real time gesture recognition using eigenspace from multi-input image sequences," presented at Automatic Face and Gesture Recognition, 1998. Proceedings. Third IEEE International Conference on, Nara , Japan, 1998.
- [18] M. C. Su, "A fuzzy rule-based approach to spatio-temporal hand gesture recognition," *IEEE Transactions on Systems Man and Cybernetics Part C-Applications and Reviews*, vol. 30, pp. 276-281, 2000.
- [19] K. Oka, Y. Sato, and H. Koike, "Real-time tracking of multiple fingertips and gesture recognition for augmented desk interface systems," presented at Automatic Face and Gesture Recognition, 2002. Proceedings. Fifth IEEE International Conference on, 2002.
- [20] J. R. New, E. Hasanbelliu, and M. Aguilar, "Facilitating User Interaction with Complex Systems via Hand Gesture Recognition," presented at Proceedings of the 2003 Southeastern ACM Conference, Savannah, GA, 2003.
- [21] J.-C. Lementec and P. Bajcsy, "Recognition of arm gestures using multiple orientation sensors: gesture classification," presented at Intelligent Transportation Systems, 2004. Proceedings. The 7th International IEEE Conference on, 2004.
- [22] F. Dadgostar and A. Sarrafzadeh, "A Fast Skin Detection Algorithm for Video Sequences," presented at International Conference on Image Analysis and Recognition (ICIAR), Toronto, Canada, 2005.
- [23] F. Dadgostar, A. Sarrafzadeh, and M. J. Johnson, "An Adaptive Skin Detector for Video Sequences Based on Optical Flow Motion Features," presented at International Conference on Signal and Image Processing (SIP), Hawaii, USA, 2005.
- [24] F. Dadgostar, A. Sarrafzadeh, and S. P. Overmyer, "Face Tracking Using Mean-Shift Algorithm: A Fuzzy Approach for Boundary Detection," presented at Affective Computing and Intelligent Interaction, Beijing, China, 2005.