Vision, Video, and Graphics (2003)
P. Hall, P. Willis (Editors)

Parameter Acquisition of Geometric Primitives within Virtual Environments for Internet-Based Telerobotics

Jiacheng Tan,[1] Gordon Clapworthy[2]

[1] Department of Computer Science and Software Engineering, University of Portsmouth, Portsmouth, PO1 3HE, UK
[2] Department of Computer and Information sciences, De Montfort University, Milton Keynes, MK7 6HP, UK

Abstract

With the aim of improving the efficiency of on-line modelling of virtual environments (VEs) in Internet-based telerobotics (IBT), this paper addresses the modelling of geometric primitives using only a single view of them. By studying the properties and constraints of the primitives under perspective projection, we propose a modelling framework that uses virtual features as matching templates for the parameter acquisition of 3D objects, the analytical descriptions of which are hard to find.

Categories and Subject Descriptors (according to ACM CCS): I.3.7 [Computer Graphics]: virtual reality I.2.9 [Robotics]: operator interfaces I.2.10 [Vision and Scene Understanding]: modelling and recovery of physical attributes

1. Introduction

In scientific investigations, and in some industries, it has been not uncommon to use a teleoperated robot to negotiate an inhospitable environment. With such a system, an operator can perform tasks that would otherwise require a human operator to be "on site". Normally, these systems are used only for performing some critical investigation and are accessible only by highly trained operators.

For less critical and cost-sensitive applications, the usage of such a system tends to be restricted by many factors. Two of these are the establishment of a dedicated communication line and the use of special-purpose "gadgets". The continued advances in computer technology mean that a general-purpose PC is now suitable for supporting quite a complex system. Likewise, the emergence of the Internet as a ubiquitous communication network and continuing improvements in its speed and accessibility have opened possibilities that teleoperation systems are only just beginning to exploit.

The effect of shifting teleoperation to the Internet is far-reaching: it changes the profile of the user group, it brings telerobots into domestic and commercial applications and it pushes the Internet to a new dimension. It also changes our concepts about the design and assessment of teleoperation systems and, most of all, about the interface between system and operator. It has already been demonstrated that IBT can provide a good framework for supporting collaborative tasks and the easy relocation of operators [1], for remote manufacturing [2], and for many other applications still to come.

However, at present it would challenge an Internet-based robot (IBR) to assign it a task undertaken by a conventional teleoperation system. This is because the Internet seriously weakens the most important feedback of a teleoperation system – visual feedback – although live videos, mono or stereo, also have the limitations of their own. The bandwidth of the Internet is not adequate at present for live video transmission, especially when the content of the video is time-critical such as when supporting teleoperation. As a result, many systems reported in the literature have to rely on snapshots of the working scene to guide operations [3, 4, 5]. Inevitably, these operations have been laborious and error-prone.

As a result of experience in conventional teleoperation, VEs have been introduced into Internet-based telerobots as a technical solution to the previous problems [6, 7]. Investigations of both conventional [8] and Internet-based teleoperation have suggested that many useful control techniques, such as task preview and rehearsal, can be achieved in VEs. But a VE proves to be more valuable for IBT than for its conventional counterpart: it can reduce the level of communication between the local and remote sites, thus reducing latency and removing the major impediment to effective IBT. The local

VE can frequently be updated from a relatively small amount of information – new joint angles for the robot arm or end effector if the robot is acting autonomously, for example – which can be transmitted rapidly to enable speedy update of the local VE representing the scene. It also helps to achieve good system-user interaction by presenting an immediately viewable representation of the user's actions.

It is obvious that the extent to which a VE can support task operations depends greatly upon how well the VE can represent the physical environment, especially if that environment is changing. Unfortunately, because a VE is normally created using explicit 3D models, it is difficult to update the VE when unpredictable changes take place in the real environment at operation time. As typical work scenes of teleoperation consist of simple industrial objects, this problem can be addressed by using image-based modelling, which allows the changes to the main features of the scene to be tracked and updated manually [9].

This paper reports a further step of our research towards that direction. Our purpose has been that by developing and integrating some automatic components in the modelling loop, operator's intervention of the modelling process can be kept to the minimum level. To control the scope of the problem, we restrict our investigation to the modelling of geometric primitives that can be found in many IBT scenes. By considering modelling of these primitives, we can deal with a range of dynamic phenomena typical of IBT scenarios, such as the presence of new objects, object displacements due to external intervention, or faulty operations such as the physical manipulator accidentally dropping an object somewhere. For this, we presume that the objects involved are either, themselves, geometric primitives, such as long cylindrical bars or cuboid blocks, or can be closely approximated by a small set of such primitives.

The above matters will be addressed at a level suitable for most current IBT applications and current network technology. That is, the changes to the work environment can be captured by a camera and the relevant image can be downloaded to the user in a reasonable time. In this sense, we require that both the changes in the environment and the operation tasks are not hugely time-critical, and that the current Internet bandwidth is appropriate to support them.

2. Background and Technical Considerations

VEs are generally created from graphical models of objects which are represented by explicit coordinates of vertices, their connectivity and various other attributes. A typical modelling process is usually too complicated to be performed on-line, though some scanning systems, such as laser scanners, can support the acquisition of a 3D model on-line. However, the obvious limitations of these devices, such as their limited working volumes and the need for free navigation or different views of the work sites, prevent them from being used for capturing a large practical teleoperation site.

Under certain circumstances, methods proposed in the areas of machine vision, image processing and computer graphics can be employed to recover full or partial 3D models[10, 11, 12, 13, 14]. However, for the systems under consideration in this paper, it is difficult to sustain the requisite conditions for these methods to work [15].

Considering the restrictions on an IBT system, a modelling method based on the minimum available information, a single image, had been proposed [9]. This method, based on camera calibration and image and workspace analysis, provides a feasible way of dealing with a fairly wide range of dynamic phenomena within teleoperation. By editing the VE, the method allows a typical teleoperation scene to be constructed at run-time, thus providing the freedom of migration of the teleoperation system to different worksite settings. By updating the VE, synchronisation between the physical and the virtual environments can be achieved, which guarantees that the VE provides up-to-date information about the dynamic objects for subsequent operations. Compared with approaches such as image augmentation [16], the method is more suitable and computationally more economical for this particular application.

Our system extends the ideas presented in this system; it was influenced by two main factors. Firstly, single-image-based methods often rely heavily on the user's skill in acquiring model parameters – that is, the orientation, location and dimension of an object – which tends to produce unpredictable performance. Secondly, we notice that, in most cases, the changes to the environment are in the form of displacements of existing objects. For these existing objects, whether originally modelled on-line or off-line, we keep a registration of their model parameters and locations in 3D space, so we are required, in most cases, only to track their motions rather than to re-model them.

Thus, it is worthwhile investigating an automatic, or semi-automatic, parameter-acquisition and tracking method. Inspired by observations of human behaviour when performing feature matching, and by the image-feature detection of robotic vision research, we propose a searching method.

We consider typical manual parameter-acquisition processes – the operator is required to pick and digest cues within the image and to associate them with attributes of the object such as category, location and orientation. Researchers still have a very incomplete understanding of how the human brain assesses the importance of each cue and how the inter-relationships between various cues are exploited [17]. Sometimes apparently faint or insubstantial cues can prove to be vital. However, the fact that one is able to deal with the problem successfully suggests that there must be enough evidence for one to determine these attributes inter-dependently or independently. In these processes, it is obvious that the operator's previous experience and knowledge of the scene or objects has played an important role.

However, the incomplete understanding of the issue and the lack of adequate tools for modelling the experiences or knowledge prevent us from developing a complete, powerful knowledge model in terms of defining the inter-relationship between different features, let alone defining an object model as a synthesis of the features. This consideration hinders us from pursuing a rigorous analytical approach to image-feature synthesis. Instead, it encourages us to investigate a framework that looks at the problem at a higher level - at the object level. Following this idea, we seek a technique that is built upon consideration of the 3D entities as a whole; thus, overall information of an entity, rather than fragments of it, is used for the purposes of parameter identification.

Even so, we notice that parameter acquisition by detecting features as 3D entities from a single image seems rarely to have been investigated in computer and machine vision. The most successful example in this regard is probably the detection of spheres. Although various methods and theories for recognising more complex 3D objects from a single image [18, 19] or from other data types, such as range data [20], have been reported, we find that it is still hard to get a quantitative description of the objects from these methods. The majority of the work conducted to date has considered the detection of planar objects, for example, straight lines, planes, polygonal shapes as objects that possess a linear analytical description, or circles, ellipses and arcs as objects that possess a non-linear analytical description. The main reasons for this are that these planar objects are the basic components from which more complex objects in higher dimensions can be constructed, and that only in exceptional cases, such as for the sphere, is there a sufficiently simple analytical description of a three dimensional object for most detection methods to work.

However, for the purpose of parameter acquisition, detecting 3D features as complete entities has some advantages over breaking the 3D features down into simpler sub-features and detecting the sub-features separately. Local corruption and distortion of the image, and occlusions among the features, can frequently lead to detection errors or even false detection of individual features – the smaller the feature, the more likely it is that its detection will be compromised. In contrast, by detecting complete 3D features, more constraints tend to be introduced on the individual features. This frequently leads to an over-defined problem, which is very useful in eliminating the ambiguities and uncertainties associated with imperfections in the image.

Besides the above considerations, other reasons suggest that such a technique would be ideal for our problem. The detection problems in our application are largely concerned with parameter identification rather than object recognition, and we can regard the models for the object to be detected as available, at either an actual or a conceptual level. If the purpose of detection is to track an object, for the purpose of graphics rendering, we must have a quantitative description of the object at first place; if we are modelling a new (unregistered) object, we shall leave the human operators to produce a conceptual model of the object because that is still the most reliable and efficient solution to the problem. Thus, it may not be very favourable for us to analyse very low-level features and infer from them the 3D models. Our concerns are with the orientation, the location and the geometric parameters of the models.

Considering all of these factors, we propose a method similar to that used to detect planar corners, in which templates are used to scan the image; the scanning algorithm then produces a large response at the position of potential corners. However, instead of using the templates to pick out the basic image features that will enable the 3D models to be constructed, we use the projection of virtual objects as masks and scan the parameter space of the virtual object to find the best match between the features of the image and those of the virtual object. In other words, we have a virtual object that is a potential match for an object in the scene; we seek to identify how this virtual object should be oriented, etc., to create a good match with the form portrayed in the image. In this sense, our success in acquiring some of these parameters relies to a large extent on our ability to achieve a perfect matching or alignment of the models with the image of the real objects.

Further, we notice that when creating a image of a 3D object by using the perspective projection, in most cases it is not possible to derive an analytical relationship between this image and the parameters of the object. Hence, the implementation of the method usually requires a search. If many parameters are to be found, the method will not be efficient, so the parameter space should maintain a low dimensionality, ideally less than three. In other words, if we can reduce the number of possible states of the object that we need to consider, the search can be performed more effectively.

In the following sections, we analyse the problems of parameter acquisition of geometric primitives and present the framework implied by the above ideas. Although the method is essentially oriented to the detection of geometric primitives as 3D entities, techniques that are readily available for detecting 2D features have been extensively used to isolate the parameters and reduce the parameter space. Specific application to cuboids is described in Section 4 and to cylinders in Section 5.

In the analysis, we reasonably assume that the image from which the geometric primitives are to be recovered has been adequately processed so that descriptions about the basic features of the objects are available. For example, straight lines on the image can be detected by techniques such as the Hough transform, or methods suggested by Davies [21] for line regression, or Atiquzzaman [22] for the efficient determination of lines together with their end coordinates.

3. Parameter Acquisition for Geometric Primitives

Some objects exhibit well-defined properties under the perspective projection. A single image of the object gives the complete information of the object in space; this property can be used by many useful applications. A typical object of this category is a sphere. Its perspective image is an ellipse. The symmetry of spheres, which present no problems of self-occlusion under the perspective projection, and the easy detection of ellipses make them ideal for automatic detection.

In some orientations, the self occlusion of objects such as cylinders and cones can make important features inaccessible from a single view. Further, the high dimensionality of their parameter spaces are less appealing, although their appearances in such applications are frequent. Parameter acquisition for cylinders will be addressed in Section 5.

A more general case is of various polygonal objects. These objects are very important in the sense that more complicated objects can be constructed from them. The self occlusion of the objects proves to be a problem; a single view is generally not enough to expose their geometric properties. Meanwhile, the fact that we generally need six parameters to describe their poses (positions and orientations) and more than three parameters to describe their dimensions makes them even harder to deal with if a single view is considered. However, a measure of their basic features, such as the structures of their corners by way of pre-defined configurations or by operator's perception, allows their parameters to be fully or partially recovered.

As a foundation, we shall introduce some basic conclusions from the study of 3D point sets and edged corners under perspective projection [15].

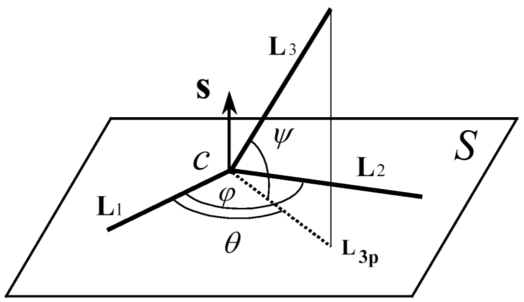

Figure 1: *A three-edge corner.*

We first consider a point set containing 3 points – these define a triangle in space, *T*. A study of the projections of the point set reveals that given the distances between the points a single view of the point set is sufficient to determine their depths if the camera is calibrated. Consequently, their orientation in the space can be determined in terms of a normal of the plane containing these points. Further, considering another 3-point set that forms a triangle T' that is similar to *T*. It is sufficient to conclude that T and T' have the same orientation, and their positions are determined up to a scale factor, if the two triangles possess the same image under perspective projection.

We next consider a generic three-edge corner consisting of three edges $\mathbf{L}_i$, i = 1..3, meeting at point c, see Fig. 1. The shape of the corner is represented by three angles: θ, φ and ψ. Under perspective projection, the orientation of the corner, **s**, is fully defined by the projections of the edges on the image plane if the parameters of the corner, θ, φ and ψ, are defined, and the knowledge of the lengths of the edges allows the depth of the corner to be recovered. This conclusion is a direct extension of the results of point-set study to the triangles formed by any two edges of the corner. It is obvious that the conclusion is extendable to corners that consist of more than three edges.

Theoretical analyses show that in both cases we can arrive at a set of equations governing the relations between the features and their images. In principle, we can solve the orientations and depths of the features from these equations, but a study of these equations reveals that their solutions require a numerical technique, which cannot guarantee a solution due to the duality of the solutions and the tolerance of the equations to errors in feature measurement. Therefore, these results are more useful in helping us to gain a qualitative insight into the problems than in achieving quantitative solutions, which further encourages us to use a search-based method.

4. Parameter Acquisition for Cuboids

As mentioned above, the constraints on the basic features of polygonal objects, such as their corners, would allow their parameters to be fully or partially recovered. To illustrate this, and to control the scale of the problem, we discuss the parameter acquisition of cuboids as an example. The same principle will be applicable to more general polygonal objects, but we shall need to seek different techniques to reduce the dimensions of their parameter spaces and to eliminate ambiguities so that a parameter search is practical.

Suppose that we need to identify the parameters of a cuboid from a single view. A pre-requisite is that we need know that it *is* a cuboid. To be a cuboid requires that its corners must satisfy these constraints: $\varphi = \pi/2$, $\psi = \pi/2$, and θ is undefined and redundant (see Fig. 1). These constraints form a geometric invariant of the object. For the object to be fully modelled in a VE, we need three parameters for its dimensions, three for its position, and three for its orientation, of which only two are independent.

We choose one of its visible corners and construct a virtual corner with same property, that is, the edges of the corner are mutually orthogonal. From the conclusions on the three-edge-corner problem, we know that if we choose the position and orientation of the virtual corner so that it overlaps the image of the corner of the cuboid, then the orientation of the virtual corner is the same as the orientation of the real corner. This implies that the orientation of the corner can be obtained by searching the parameter space of the orientation. This is advantageous, especially when the image of the object is hard to describe due to noisy data or image distortion, and is necessary if the complexity of the geometry makes it impossible to formulate an analytical solution.

Overall, the problem is 7-dimensional, which is too high for a searching technique to be applied efficiently. Consequently, it is desirable to disassemble the problem into a number of searching problems in lower dimensions, preferably less than three. We show that this is possible, and in fact, that only the acquisition of orientation parameters requires a searching scheme. We carry out the process in three stages.

In the first stage, we need to gather from the image information about lines, their end coordinates and their intersections. Based on the results, it would be straightforward to find the position (up to a range factor) once the intersection of the edges of the actual corner has been detected from the image. Although the information about the position is incomplete at this stage, it provides a crucial basis on which a simpler searching scheme for orientation can be built.

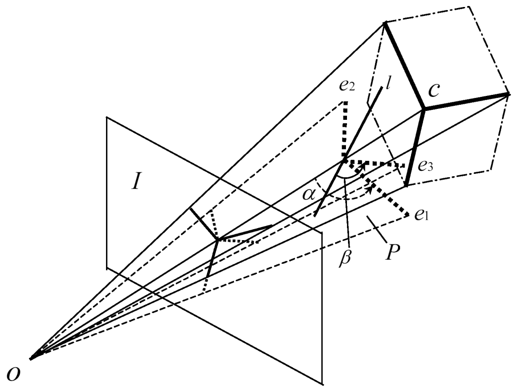

Figure 2: *Search for the orientation of a cuboid.*

Now, we move on to the second stage, to find the orientation of the corner. As the orientation of the corner is independent on the dimensions of the cuboid, we can consider the orientation problem independently. Since it is difficult to solve the orientation from the image of the corner (Section 3), a searching scheme is needed. Of the three parameters for orientation, only two are independent, so the searching problem is two-dimensional. Even so, we need a scheme to keep the searching within a two-dimensional parameter space. We can realise this by adopting a new expression for the orientation of the corner.

Suppose that, as a result of the first stage of parameter acquisition, the virtual corner has already been positioned on the projection line of the actual corner, as shown in Fig. 2. Here, the corner of the object is at c and the image of the corner on I is shown. The virtual corner, consisting of the three edges $\mathbf{e}_i$, $i = 1..3$, has been placed in alignment with the image of c.

Choose any of the three edges as an axis and rotate the virtual corner around this axis until one of its edges is on one of the planes that contain both an edge of the actual corner and the centre of projection ($\mathbf{e}_1$ and P, respectively, in Fig. 2). If we subsequently restrict the motion of $\mathbf{e}_1$ so that it continues to lie on P, then the orientation of the virtual corner can be described by two angles: α and β. The angle α is defined as the angle between $\mathbf{e}_1$ and the line connecting O and c. The angle β is defined as the angle between one of the two remaining edges of the virtual corner (that is, $\mathbf{e}_2$ or $\mathbf{e}_3$) and the line l that is the intersection of the plane P and the plane defined by $\mathbf{e}_2$ and $\mathbf{e}_3$. The angle β represents the rotation of the virtual corner about $\mathbf{e}_1$.

According to the scheme, it appears that the searching ranges would be $(-\pi, \pi)$ for α and $(0, 2\pi)$ for β. In practice, however, the ranges for both α and β are restricted by the patterns of the image of the actual corner. The restrictions exist because we require that the virtual corner remains convex and that all its edges must be visible, otherwise we do not have sufficient constraints to solve the problem. Fig. 3 shows the patterns of the image of the corner and the corresponding ranges for α and β.

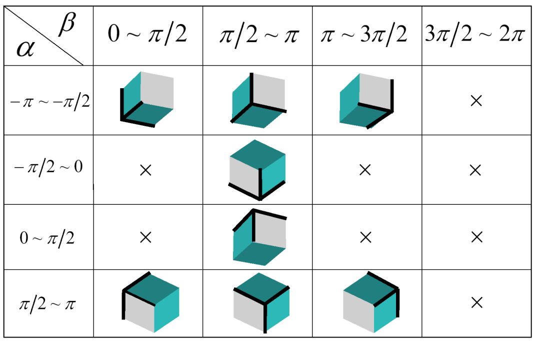

Figure 3: *Possible patterns of convex corners.*

Once the orientation of the corner has been identified, we are able to complete the process by back-projecting the end coordinates of the edges, which has been acquired at the first stage, on to the corresponding edges of the virtual corner. Together with the vertex, the newly found points define a unique virtual cuboid. At this stage, besides the edges of the corner under consideration other visible or partially occluded edges and corners can be used to help to achieve a good estimation of the dimensions of the cuboid.

The virtual cuboid found in this way is different from the real cuboid by a scale factor. To determine this factor, we need to find an anchor of the object, such as a neighbouring object, the surface on which it rests, etc., as needed by the single-image-based method.

5. Parameter Acquisition for Cylinders

The principle of Section 4 is applicable also to other geometric primitives. In this section, we show how it can be applied to a cylinder and how the redundant information can be utilised to refine the parameters.

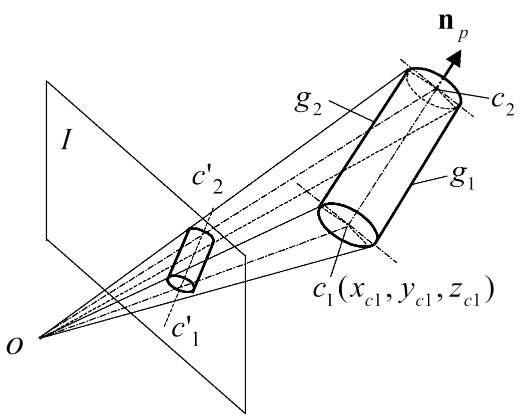

Figure 4: *A cylinder under perspective projection: the cylinder and camera system.*

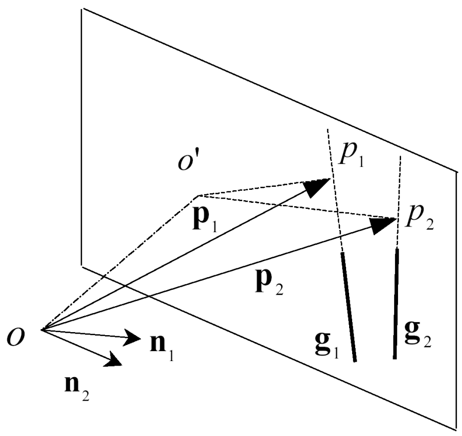

Figure 5: *The images of the cylindrical edges of the cylinder and camera system.*

A cylinder in space is described totally by nine parameters: three for its position, three for its orientation, two for its dimensions and one for its rotation around the axis of symmetry. Neglecting its rotation and one dependent variable for its orientation, and allowing an extra one for a scale factor, we still have six parameters to find. To avoid searching a high-dimensional space, the acquisition will again be conducted in stages.

First, consider a cylinder and its image, Fig. 4. Generally, we will have an ellipse-like image of one of its end circles and two line segments for its cylindrical surface, although in some cases we will not be able see either of the end circles. We can find these features by processing the image.

The image of the end circle, an ellipse, carries a lot of information about the cylinder: its position (up to a scale factor), its orientation and its radius. However, considered that the centre of the circle has not been preserved under perspective projection, this feature may not be a good primary one from which to derive the parameters such as those of position and orientation, although it provides good estimation to them, such as the diameter of the cylinder. As we shall see, other easy-to-detect features can be used for such purposes.

To do so, we choose the vector that is aligned with the axis of symmetry of the cylinder as its orientation vector, $\mathbf{n}_p$, and the centre of one of the bottom circles, c_1 , as its position, as shown in Fig. 4. Now, consider the image of the cylindrical surface. On the image plane, this surface is represented by two straight lines, $\mathbf{g}_1$ and $\mathbf{g}_2$, as shown in Fig. 5. Suppose that p_1 and p_2 are, respectively, the points at which the lines from the centre of the image intersect $\mathbf{g}_1$ and $\mathbf{g}_2$ at right angles, as can be found by the foot-of-normal method, then p_1 and p_2 define two vectors, $\mathbf{p}_1$ and $\mathbf{p}_2$. Under this convention, we have two normals: $\mathbf{n}_1$ is the normal of the plane containing $\mathbf{g}_1$ and $\mathbf{p}_1$, and $\mathbf{n}_2$ is the normal of the plane containing $\mathbf{p}_2$ and $\mathbf{g}_2$, i.e.,

$$\mathbf{n}_1 = \mathbf{p}_1 \times \mathbf{g}_1 \qquad \mathbf{n}_2 = \mathbf{p}_2 \times \mathbf{g}_2 \tag{1}$$

We know that these two planes are tangential to the cylindrical surface and that they must pass through the centre of projection of the camera. So, the line of intersection of the two planes is a line that passes through the centre of projection and is parallel to the axis of symmetry of the cylinder. Hence, we have

$$\mathbf{n}_p = \mathbf{n}_1 \times \mathbf{n}_2 = (\mathbf{p}_1 \times \mathbf{g}_1) \times (\mathbf{p}_2 \times \mathbf{g}_2) \tag{2}$$

Eq. (2) demonstrates that the orientation of the cylinder is fully determined by the perspective projections of two straight lines on the cylindrical surface. Meanwhile, these lines restrict the position of the cylinder to a plane that passes through the axis of symmetry of the cylinder and the centre of projection, O, thus reducing the number of position parameters to one (the other one accounts for the scale factor).

Now, we are ready to apply a searching technique to the remaining parameters – a three-dimensional search problem. This is not a laborious process if we note that the images of bottom circles and the cylindrical surface can provide very good estimations to the position, the radius and the length of the cylinder, even if they are not good enough to be used as primary features.

It is clear that, as mentioned, the presence of the redundant constraints is advantageous, and taking the seemingly redundant constraints into account can usually lead to a better estimation of the model of the object. Meanwhile, the use of a 3D virtual object as a matching template makes all the possible relationships between different features to be employed in a consistent way, which makes the method robust to noises and occlusions.

6. Conclusions

By studying the constraints of geometric primitives under perspective projection, and by using their 2D features in the image, we have been able to develop a method that uses 3D virtual objects as searching templates to acquire the parameters of geometric primitives. Using 3D virtual object allows all the possible relationships between different features to be employed in a consistent way, which makes the modelling process more robust to noise in the image.

The results provide a framework within which the performance of the modelling process of VE can be improved by introducing automatic components into the parameter identification if a requirement for online modelling emerges when a dynamic environment is involved in teleoperation.

From this work, we can identify several directions for further investigations. One aspect of high priority is to combine the method with techniques that can analyse and trace the references (lines/surfaces) from the neighbouring objects so that the scale factors of the primitives being modelled can be automatically determined.

References

1. P.G. Backes, K.S. Tso and G.K. Tharp, "Mars pathfinder mission Internet-based operation using WITS", *Proc. IEEE Int. Conf. Robotics and Automation*, Leuven, Belgium, pp. 284–291 (May 1998). 1
2. R.C. Lou, W.Z. Lee, J.H. Chou and H.T. Leong, "Tele-control of rapid prototyping machine via Internet for automated tele-manufacturing", *Proc. IEEE Int. Conf.Robotics and Automation*, Detroit, Michigan, pp. 2203–2208 (May 1999). 1
3. K. Goldberg, M. Mascha, S. Getner and N. Rothenberg, "Desktop teleoperation via the World Wide Web", *IEEE Int. Conf. on Robotics and Automation*, Nagoya, Japan (May 1995). 1
4. K. Taylor and J. Trevelyan, "Australia's telerobot on the Web", *Proc. 26th Int. Symposium on Industrial Robots*, Singapore (October 1995). 1
5. M.R. Stein, "Painting on the World Wide Web: the PumaPaint project", *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, Victoria, Canada (October 1998). 1
6. I.R. Belousov, J. Tan and G.J. Clapworthy, "Teleoperation and Java3D visualisation of a robot manipulator over the World Wide Web", *Proc. IEEE Int. Conf. Information Visualization '99*, London, UK, pp. 543–548, (July 1999). 1
7. I.R. Belousov, R. Chellali and G.J. Clapworthy, "Virtual reality tools for Internet robotics", *Proc. IEEE Int. Conf. Robotics and Automation*, Seoul, South Korea, 2001. 1
8. A.K. Bejczy, "Virtual reality in telerobotics", *Proc. IEEE Int. Conf. Robotics and Automation*, Saint Feliu de Guixols, Spain (September 1995). 1
9. J. Tan, G.J. Clapworthy and I.R. Belousov, "Single-image-based on-line camera calibration and VE modeling method for teleoperation via the Internet", *Proc. SPIE*, vol. 4302, pp.1–12, 2001. 2
10. J.T. Camillo and J.K. David, "Structure and motion from line segments in multiple images", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, **17**(11):1021–1032, 1995. 2
11. K. Nagao and W.E.L. Grimson, "Object recognition by alignment using invariant projections of planar surfaces", *Proc. Int. Conf. Pattern Recognition*, pp.861–864, 1994. 2
12. T. Vetter and T. Poggio, "Linear object classes and image synthesis from a single example image", *MIT A.I. Memo.*, No.1531, 1995. 2
13. P.E. Debevec, C.J. Taylor and J. Malik, "Modeling and rending architecture from photographs: a hybrid geometry- and image-based approach", *Proc. SIGGRAPH'96*, pp.11–20, 1996. 2
14. Y. Horry, K. Anjyo and K. Arai, "Tour into the picture: using a spidery mesh interface to make animation from a single image", *Proc. SIGGRAPH'97*, Los Angeles, California, pp.225–232 (August 1997). 2
15. J. Tan, "The use of virtual environments in Internet-based teleoperation", Ph.D. thesis, De Montfort Univ. Milton Keynes, U.K., 2001. 2, 4
16. A. Rastogi, P. Milgram and D. Drascic, "Telerobotic control with stereoscopic augmented reality", *Proc. SPIE*, vol. 2653, pp.115–122, 1996. 2
17. R. Sekuler and R. Blake. *Perception*. McGrw-Hill Inc., 1994. 2
18. D.G. Lowe, "Three-dimensional object recognition from single two-dimensional images", *Artificial Intelligence*, **31**(3):355–395, 1987. 3

19. I. Biederman, "Recognition-by-components: a theory of human image understanding", *Psychological Review*, **94**:115–147, 1987. 3

20. R.C. Bolles and R. Horaud, "3DPO: a three-dimensional part orientation system", *Int. J. Robot. Res.*, **5**(3):3–26, 1986. 3

21. E.R. Davies, "Image Space Transforms for Detecting Straight Edges in Industrial Images", *Pattern Recognit. Lett.*, **4**(2):185–192, 1986. 3

22. M. Atiquzzaman and M.W. Akhtar, "Complete line segment description using the Hough transform", *Image and Vision Comp.*, **12**(5):267–273, 1994. 3