A Note on Hybrid Control of Robotic Spatial Augmented Reality

Joo-Haeng Lee[1], Junho Kim[2] and Hyun Kim[3]
[1,3] Robot & Cognitive System Research Department, ETRI, Daejeon, 305-700, Korea
(Tel : +82-42-860-1338; E-mail: {joohaeng | hyunkim}@etri.re.kr)
[2] School of Computer Science, Kookmin University, Seoul 136-702, Korea
(Tel: +82-2-910-4826; E-mail: junho@kookmin.ac.kr)

Abstract - A robotic spatial augmented reality (RSAR) system combines robotics with spatial augmented reality (SAR) where cameras are used to recognize real objects, and projectors augment information and user interface directly on the surface of the real objects, rather than relying on mobile or wearable display devices. Hence, the control of a RSAR system requires handling of different types of control schemes at once such as classical inverse kinematics of simply linked bodies, inverse projections to find appropriate internal/external parameters of a projector, and geometric manipulation of a projection source image to increase the flexibility in control. In this paper, we outline a hybrid approach to control relevant control components in a coordinated manner, specially focused on application in a prototype RSAR system developed in ETRI.

Keywords – inverse projection, inverse kinematics, image manipulation, spatial augmented reality, robotic computer

1. Introduction

Spatial Augmented Reality (SAR) is a relatively new branch of augmented reality (AR) [1,2]. In SAR, we augment information and user interface directly on a surface of physical objects rather than using an explicit mobile display between a user and the physical world [3,4]. In particular, recent development of mobile projectors has accelerated this research field [5-15]. Another key enabler is a consumer depth camera (i.e., Kinect) to easily get the geometry of environment and objects [12-14].

For an interactive SAR application, we need to determine both internal and external parameters of a projector as well as to manipulate a source image to achieve the required target image with a specified shape and location considering a user's viewpoint. Moreover, when a projector is mounted on a robotic platform with several extra DOFs as in a *Robotic Spatial Augmented Reality* (RSAR) system [9,10,11], the control complexity rapidly increases due to the emergence of the new type of end-effector and abundant DOFs.

Generally, the control of a RSAR system requires handling of different types of control schemes at once such as classical inverse kinematics of simply linked bodies, inverse projections to find appropriate internal/external parameters of a projector, and geometric manipulation of a projection source image to increase the flexibility in control. In this paper, we outline a hybrid approach to control relevant control components in a coordinated manner, specially focused on application in a prototype RSAR system developed in ETRI [16,17,18].

The rest of this paper is organized as follows. In section 2, we explain three major control components: inverse kinematics, inverse projection, and image manipulation. When explaining each component, we give corresponding examples resulted from the computer simulation of related algorithms. In section 3, we outline our hybrid approach to handle these control components in a coordinated manner with an example. Finally, we conclude this paper with remarks on future research directions.

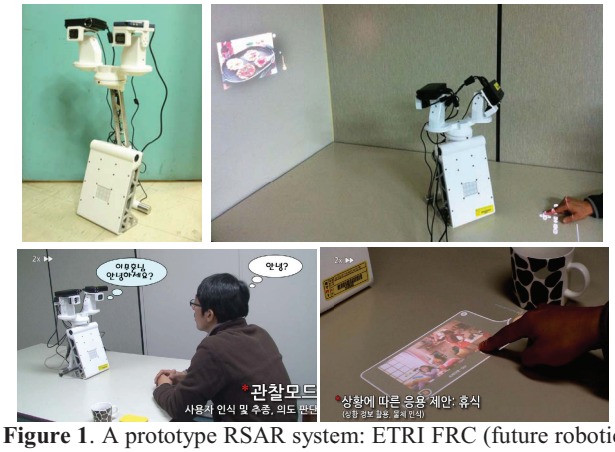


Figure 1. A prototype RSAR system: ETRI FRC (future robotic computer): (top left) working prototype; (top right) dual projectors for information on the wall and user interface on the table; (bottom left) multi-modal interaction including voice and face recognition; (bottom right) augmentation of user interface around a physical object, in this case a coffee cup.

2. Control Components

In this section, we describe characteristics and basic formulation of kinematics, projection, and image manipulation process, specially focused on an experimental RSAR system being developed in ETRI [16]. Table 1 summarizes notations and abbreviations used in next sections.

2.1 Inverse Kinematics

A. Kinematic Setup

Our experimental system, ETRI FRC (Future Robotic Computer) has 5 motors: two pairs of pen-tilt arms are mounted on a common base pan motor, which can extends the operation limits of each arm pan while causing redundancy in control [17]. The kinematic configuration of FRC is represented as follows:

978-1-4577-0723-0 / 11 / $26.00 ⓒ 2011 IEEE

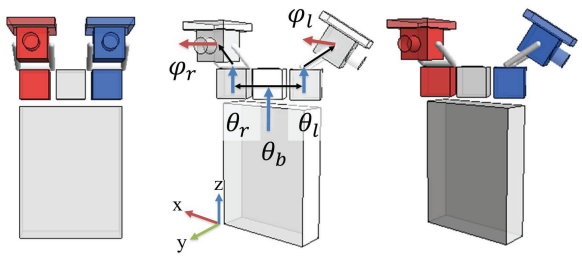
Authorized licensed use limited to: University of Calgary. Downloaded on August 13,2021 at 18:36:22 UTC from IEEE Xplore. Restrictions apply.

$$k = (\theta_b; \theta_l, \varphi_l; \theta_r, \varphi_r) \quad (1)$$

where θ and φ and denote pan and tilt angles, respectively. And, subscripts b, l and r denote base, left, and right modules, respectively. (See Figure 2.)

Table 1.Nomenclatures and Abbreviations

Notations	Meaning
k	A kinematic configuration
q	A kinematic end-effector as an output of a forward kinematics function A; Otherwise, a projector configuration as an input to a forward projection function B.
p	A projection target, usually a planar convex quadrilateral.
p_x	A planar convex quadrilateral whose two diagonals intersect at the point x.
CQ	A convex quadrilateral whose projectability is not known yet.
PCQ	A projectable convex quadrilateral
NPCQ	A non-projectable convex quadrilateral



(a) (b) (c)

Figure 2. Kinematic schematics of FRC [17]: (a) A canonical posture. The top most attachment is a mobile projector. A cylinder represents a camera. Each cube represents a motor. (b) Kinematic parameters of Eq. (1), (c) a modified posture.

When a RSAR system has mobility, its DOF can be extended as

$$k_m = (\theta_b; \theta_l, \varphi_l; \theta_r, \varphi_r; \theta_m, p_m) \quad (2)$$

where $\theta_m \in R$ and $p_m \in R^3$ denote pan-wise orientation and translational displacement, respectively, of the local frame due to mobility. In practice, mobility parameters can be handled independently with the other parameters. In this paper, we focus on the case without mobility.

On each arm tip, a projector-camera pair is mounted and geometrically calibrated [19]. We assume that the kinematic end-effector is defined with two components: (1) the center of the projection $c_i = (x_{c,i}, y_{c,i}, z_{c,i}) \in R^3$ and (2) the direction of principal axis $d_i = (\theta_{p,i}, \varphi_{p,i}) \in R^2$ for each projector $i \in \{l, r\}$. Note that, in RSAR, the final end-effector is represented as a target projection area as in Section 2.2.

B. Forward Kinematics

For a given kinematic configuration k of Eq. (1), let A be a forward kinematics function to compute the kinematic end-effector:

$$A(k) = A(\theta_b; \theta_l, \varphi_l; \theta_r, \varphi_r) = (c_l, d_l; c_r, d_r) = (q_l, q_r). \quad (3)$$

The function A is defined in a classical manner as a composition of the sequence of rigid body transformations. When controlling an individual projector-camera pair, it is convenient to ignore the other pair:

$$A_i(k_i) = A_i(\theta_b; \theta_l, \varphi_l) = (c_i, d_i) = q_i, \quad (4)$$

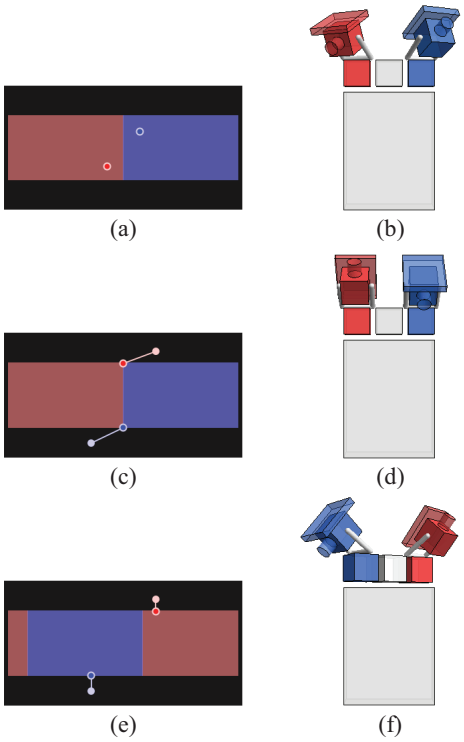
where $i \in \{l, r\}$.

C. Inverse Kinematics

When kinematic end-effectors (q_l, q_r) are given, we can find a configuration k that best approximates the given end effector:

$$A^{-1}(q_l, q_r) = (\theta_b; \theta_l, \varphi_l; \theta_r, \varphi_r) = k, \quad (5)$$
$$A(k) = (\hat{q}_l, \hat{q}_r) \approx (q_l, q_r),$$
$$A^{-1}(q_i) = (\theta_b; \theta_i, \varphi_i) = k_i,$$
$$A(k_i) = \hat{q}_i \approx q_i,$$

In a simple structure, the inverse process of Eq. (5) can be computed analytically while satisfying the kinematic constraints of a RSAR system. In a general case, we can apply differential inverse kinematics or Jacobian Transpose methods []. Figure 3 shows an example of inverse kinematics control where analytic approach was applied to find the pair of directional end-effectors [17].



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

Figure 3. Examples of inverse kinematic control [17]: the left column shows the target end-effectors $(\boldsymbol{q_l}, \boldsymbol{q_r})$ composed of direction of principal axis $\boldsymbol{d_i} = (\boldsymbol{\theta_{p,i}}, \boldsymbol{\varphi_{p,i}})$, and the right column shows the corresponding postures generated with kinematic configurations $\boldsymbol{k = A^{-1}(q_l, q_r) = (\theta_b; \theta_l, \varphi_l; \theta_r, \varphi_r)}$. The solid red and blue dots denote target directions in the pan-tilt domain, and light dots denote approximate solution within the kinematic constraints. The first row is the canonical posture. The second row is the case when base pan is fixed. The third row is the case when the base pan redundancy is used to better satisfy the goal.

Authorized licensed use limited to: University of Calgary. Downloaded on August 13,2021 at 18:36:22 UTC from IEEE Xplore. Restrictions apply.

2.2 Inverse Projection

A. Projection Setup

A configuration of a projector is represented as a collection of external and internal parameters:

$$q = (x, y, z; \theta, \varphi; f, a) = (c, d; f, a) \quad (6)$$

where $c = (x, y, z) \in R^3$ is the location of the center of the projection, $d = (\theta, \varphi) \in R^2$ is the direction of the principal axis, f is the focal parameter (or throw rate), and a is aspect ratio of the source image. (See Figure 4.) We can omit internal parameters f and a from a configuration when they are fixed in a system.

$$q = (x, y, z; \theta, \varphi) = (c, d). \quad (7)$$

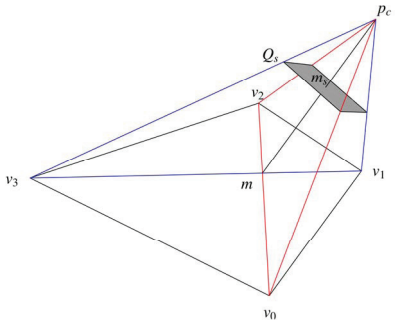


Figure 4. Projection schematics [18] defined in a frustum whose apex is the center of projection $\boldsymbol{c}$. The direction $\boldsymbol{d}$ of principal axis is defined along a line passing $\boldsymbol{c}$ and the projected mid-point $\boldsymbol{m}$. The projection source is gray rectangle normal to *d*, and the target quadrilateral is the lower face of the frustum.

An end-effector of a projector (or a RSAR system) can be described as a projection area in a physical world. Since we assume a flat world in this paper, it is a planar convex quadrilateral. A planar quadrilateral target image is described with four vertices:

$$p = (p_0, p_1, p_2, p_3), \quad (8)$$

where $p_i \in R^3$ denotes the *i*-th vertex of a quadrilateral p.

B. Forward Projection

Forward projection can be formulated with a transformation function B that maps a given configuration q to a projected area, which is a convex quadrilateral $B(q)$ on a planar surface s:

$$B(q) = p = (p_0, p_1, p_2, p_3). \quad (9)$$

When target surface is a plane, forward projection B can be simply computed using a projective geometry, specially using homography relation [20]. Otherwise, we have to compute intersection between a target surface and a projector frustum, which may be costly and non-trivial for a free-form surface.

C. Diagonal Parameterization of Convex Quadrilaterals

In this section, we review a relevant method to transform a convex quadrilateral into its canonical form based on *diagonal parameterization*, which is required to apply a known inverse projection method [18].

Let C be a function to transform an arbitrary convex quadrilateral p in space into its canonical form $\tilde{p} = C(p)$ on a plane of normal (0,0,1) with following affine transformation: (1) translation: the intersection of the diagonals are at the origin; (2) rotation: the shorter diagonal is aligned along x-axis; and (3) scale: the length of shorter diagonal is normalized to unit length.

$$C(p) = (\tilde{p}; \alpha). \quad (10)$$

When the transformation parameters α is provided with a canonical convex quadrilateral $\tilde{p}$, its original form p can be reconstructed:

$$p = C_\alpha^{-1}(\tilde{p}). \quad (11)$$

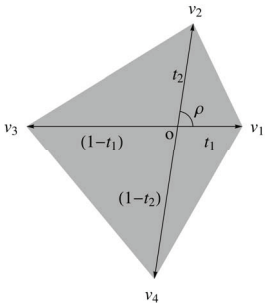


Figure 5. Diagonal parameterization of a convex quadrilateral: $(\boldsymbol{l_2}, \boldsymbol{\rho}, \boldsymbol{t_1}, \boldsymbol{t_2})$.

A canonical convex quadrilateral can be uniquely parameterized with four parameters (l_2, ρ, t_1, t_2) using the diagonal parameterization [18]:

$$D(\tilde{p}) = D \circ C(p) = (l_2, \rho, t_1, t_2) \quad (12)$$

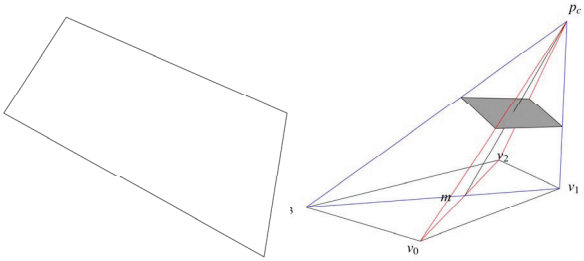
where l_2 is the length of the longer diagonal (assuming the shorter diagonal has the unit length), ρ is the angle between two diagonals, and t_i is the division ratio of the *i*-th diagonal. (See Figure 5.)

D. Inverse Projection

Recently, Lee [18] has shown that the projectability of a canonical convex quadrilateral $\tilde{p}$ can be determined by solving a pair of non-linear equations formulated with diagonal parameters assuming that a rectangle source image of an unknown aspect ratio is projected. In addition, this algorithm can find the internal and external parameters of projection, if a solution exists.

The overall process of [18] can be formulated as an inverse projection process to find a corresponding configuration $\tilde{q}$:

$$B_L^{-1}(\tilde{p}) = B_L^{-1} \circ C(p) = \tilde{q}. \quad (13)$$



(a) Target quadrilateral (b) Projector configuration found.

Figure 6. An example of inverse perspective projection.

Authorized licensed use limited to: University of Calgary. Downloaded on August 13,2021 at 18:36:22 UTC from IEEE Xplore. Restrictions apply.

Note that the projector configuration $\tilde{q}$ of Eq. (13) is computed for the canonical convex quadrilateral $\tilde{p}$. To get the actual configuration q, we need to apply the same affine transformation of Eq. (11) to $\tilde{q}$. Now, we can define the final inverse projection formulation as follows:

$$B^{-1}(p) \stackrel{\text{def}}{=} C_\alpha^{-1} \circ B_L^{-1} \circ C(p) = C_\alpha^{-1}(\tilde{q}) = q \quad (14)$$

Figure 6 shows an example of inverse perspective problem: (a) the given target quadrilateral which is PCQ, and (b) computed external parameters. The internals parameter is implicitly expressed in the aspect ratio of the source rectangle, which is 5.065 in this example.

2.3 Manipulation of Source Images

Unlike pure mechanical setup, our end-effector is affected by another source of DOF: manipulation of source image. For example, when the target image is not projectable due to kinematic constraints or not compatible in aspect ratio due a projector setup, we can change the source image itself as the third control scheme. For example, when a given target quad is not a PCQ, we may find an enclosing PCQ of the given source image.

The general expansion scheme can be formulated as follows for a given target quadrilateral p:

$$\begin{aligned} &B^{-1}(p) = q, \\ &E(q) = q_e, \\ &B(q_e) = p_e, \end{aligned} \quad (15)$$

where q_e is a new projector configuration that satisfies the expansion constraint $B(q_e) = p_e \supset p$. The specific expansion scheme depends on applications. In the below, we give two examples.

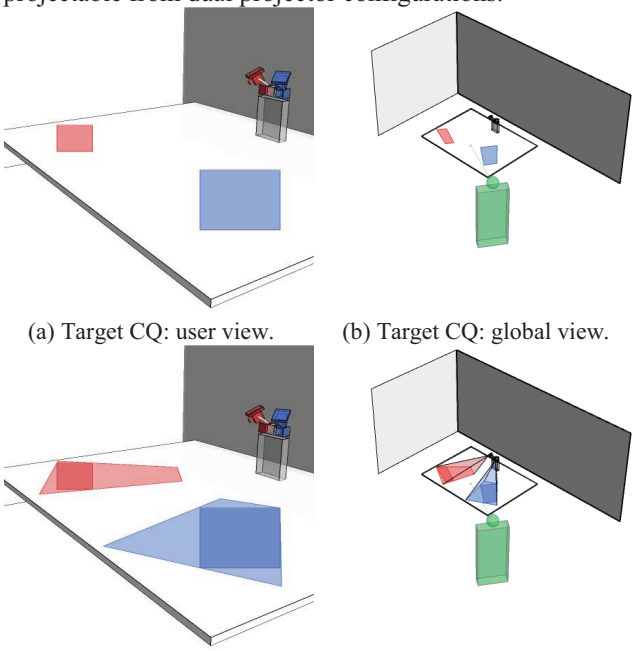
A. Expansion by Aspect Ratio

Even when the given target image is determined as projectable as in Section 2.2, the related internal parameters may not be compatible with the system. For example, when the computed aspect ratio is not compatible with the projector in the system, it is straightforward to expand the source image (usually into a single dimension) to achieve the system aspect ratio. Then, we need to change either the throw rate using a zoom lens or kinetic parameters by moving back along the direction of principal axis. Both of these schemes are coupled with system constraints, which can be handled in inverse kinematics control in Section 2.1.

B. Enclosing PCQ

An enclosing PCQ of a NPCQ is another expansion scheme with more general usage. For example, when a given CQ p does not generate a proper projector/kinematic configuration, a PCQ p_e that encloses a target CQ can be chosen while keeping the kinematic constraints. Then we can apply the inverse projection transformation to p_e rather than p.

Figure 7 shows an example where this method is used to find manipulated source images in anamorphic illusion [10]. In Figure 7a-b, the target images are special quadrilaterals which are rectangles from the user's point of view. However, since these target images are not directly projectable, we need to find enclosing PCQ that are both projectable from dual projector configurations.



(a) Target CQ: user view. (b) Target CQ: global view.

(c) Enclosing PCQ: user view. (d) Enclosing PCQ: global view.

Figure 7. An example of enclosing PCQ approach: the target images are the rectangles from the view point of a user, which is a typical case in anamorphic illusion display [10].

3. Hybrid Control

3.1 Control Flows

In the below, we summarize the unified control flows composed of three control schemes explained in Section 2.

A. Forward Control Flow

Forward: (1) a kinematic configuration $k \rightarrow$ (2) a projector configuration $q \rightarrow$ (3) a target convex quadrilateral p

For a single projector-camera pair, the forward process is straight forward using Eqs. (4) and (9):

$$B \circ A(k) = B(q) = p, \quad (16)$$

It is not usual in a forward control to geometrically manipulate a source image. If required, however, the forward process can be extended using Eq. (15) as follows:

$$B \circ E \circ A(k) = B(q_e) = p_e, \quad (17)$$

B. Inverse Control Flow

Inverse: (1) a target convex quadrilateral $p \rightarrow$ (2) a projector configuration $q \rightarrow$ (3) a kinematic configuration k

For a single projector-camera pair, the inverse process is represented as follows using Eqs. (5) and (14):

$$A^{-1} \circ B^{-1}(p) = A^{-1}(q) = k, \quad (18)$$

Extended Inverse: (1) a target convex quadrilateral $p \rightarrow$ (2) a projector configuration $q \rightarrow$ (3) an expanded projector configuration $q_e \rightarrow$ (4) a kinematic configuration k_e.

Authorized licensed use limited to: University of Calgary. Downloaded on August 13,2021 at 18:36:22 UTC from IEEE Xplore. Restrictions apply.

The above extended inverse process can be formulated using Eq. (15):

$$A^{-1} \circ E \circ B^{-1}(p) = A^{-1} \circ E(q) = A^{-1}(q_e) = k_e. \quad (19)$$

In practice, we need criteria to determine if image manipulation is required in the above process.

3.2 One-Parameter Family of PCQ

A one-parameter family (OPF) of PCQ is a continuous set of projectable convex quadrilaterals. A simple example of a one-parameter family of PCQ is generated when a projector projects a source image on a flat surface while moving smoothly. (See Figure 9.) By applying a canonical transformation to a one-parameter of PCQ, we can get a one-parameter family of canonical PCQ. In the below, we describe this concept as an example of hybrid control to determine *initial projectability test* in our prototype RSAR system.

A. Simple Case: No Redundancy in DOF

Assume that a certain RSAR system has no directional redundancy in the mechanism, and its location is fixed. In our prototype system, it can be simply achieved by fixing the base pan θ_b.

In this case, every point x on a projectable plane s can be uniquely mapped to a kinematic configuration in such a way that the principal axis of the projection frustum intersects with the plane s at the point x. We can denote this relation as $k(x) = A^{-1} \circ B^{-1}(x)$ using Eq. (20) without loss of generality. This function $k(x)$ can be analytically defined for a given mechanism.

Inversely, this parameterized kinematic configuration $k(x)$ defines a corresponding PCQ p on s as its target image in such a way that the projected mid-point of p is placed at x:

$$p_x = p(x) = B \circ A(k(x)). \quad (20)$$

Note that, in this case, every PCQ has a shape of isosceles trapezoid since there is no roll in rotation, which implies that the projectability of a given target CQ p can be easily filtered. Then, for the intersection m of the diagonals of p, we need to check if $p = p(m)$ using Eq. (20).

If p is determined as not directly projectable from the current kinematic setup, we may apply the image manipulation scheme of Section 2.3.

B. General Case: Redundancy in DOF

If there exists control redundancy as in our prototype RSAR system (due to pan motors at the base and each side), a point x on a projectable plane s is mapped to a one-parameter family of PCQ $p(x,t)$ rather than a single instance of PCQ $p(x)$, where $t \in R$ is the parameter induced from a one-parameter family of kinematic configuration $k(x,t)$.

The dimension of t depends on the level of redundancy. Generally, the parameterized configuration $k(x,t)$ is a curve or hyper-surface in a manifold, which can be analytically defined for a given mechanism.

For example in our prototype system, the base pan θ_b can be used to defined the parameterized kinematic configuration $k(x,\theta_b) = (\theta_b; \theta_r(\theta_b), \varphi_r(\theta_b))$ for a fixed surface point x. Then, the parameterized PCQ can be defined as a continuous set of varying isosceles trapezoid while the intersection of diagonals are fixed at the point x:

$$p(x,\theta_b) = B \circ A(k(x,\theta_b)). \quad (21)$$

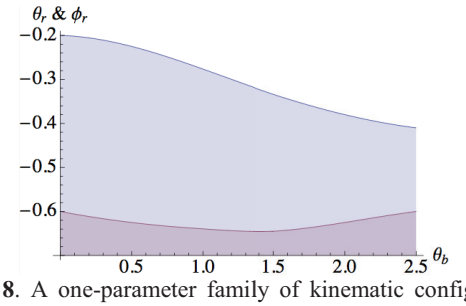
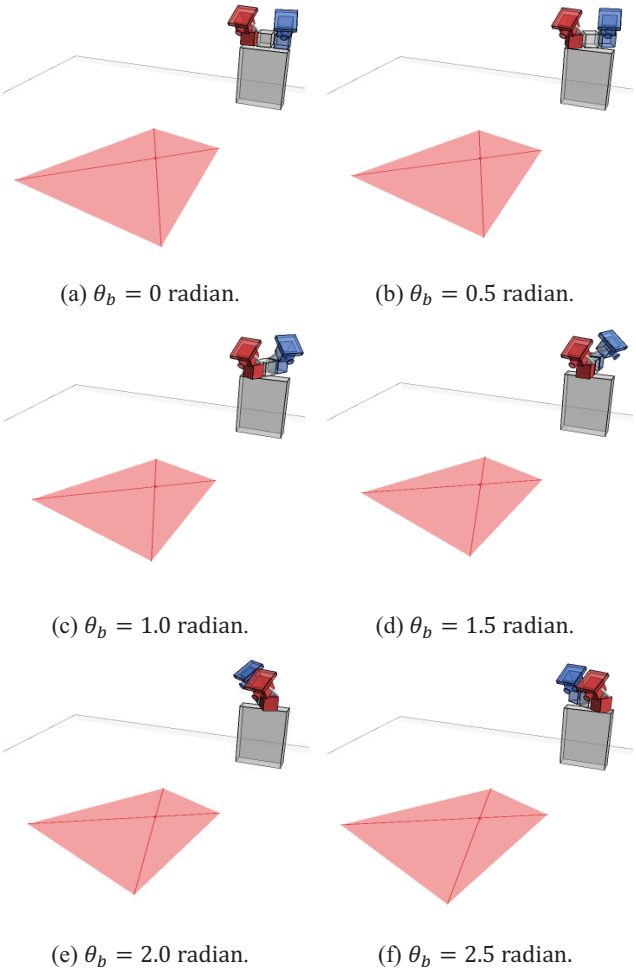


Figure 8. A one-parameter family of kinematic configuration $(\boldsymbol{\theta_b}; \boldsymbol{\theta_r}(\boldsymbol{\theta_b}), \boldsymbol{\varphi_r}(\boldsymbol{\theta_b}))$ that generates a one-parameter family of PCQ whose projected mid-point is fixed at x, which is the intersection of diagonals in Figure 9.



(a) $\theta_b = 0$ radian. (b) $\theta_b = 0.5$ radian.

(c) $\theta_b = 1.0$ radian. (d) $\theta_b = 1.5$ radian.

(e) $\theta_b = 2.0$ radian. (f) $\theta_b = 2.5$ radian.

Figure 9. An example of a one-parameter family of PCQ. Each instance is generated from the varying kinematic configuration $(\boldsymbol{\theta_b}; \boldsymbol{\theta_r}(\boldsymbol{\theta_b}), \boldsymbol{\varphi_r}(\boldsymbol{\theta_b}))$ of Fig. 8 with the constraint that the intersection of diagonals is identical and the RSAR system has no mobility.

Figure 9 shows an example of a one-parameter family of PCQ for a fixed surface point: each instance of PCQ is

Authorized licensed use limited to: University of Calgary. Downloaded on August 13,2021 at 18:36:22 UTC from IEEE Xplore. Restrictions apply.

sampled from the parameterized kinematic configuration $k(x, \theta_b) = (\theta_b; \theta_r(\theta_b), \varphi_r(\theta_b))$, which is illustrated in Figure 8. Note that each sample is similar in shape, but has different diagonal parameterization.

If a given target CQ p_x belongs to the one-parameter family of PCQ $p(x, t) = B(q(x, \theta_b))$, it is inversely projectable in a geometric sense. Otherwise, we may choose an enclosing PCQ as in Section 2.3. Note that the enclosing PCQ may have the different projected mid-point $y \neq x$ as in Figure 7:

$$p_x \subset p(y, \theta_b) = B \circ A(k(y, \theta_b)). \quad (22)$$

4. Conclusion

In this paper, we have introduced a general idea of controlling a robotic spatial augmented reality (RSAR) system, specially focused on a prototype system developed in ETRI. We have incorporated different control modules such as inverse kinematics, inverse projection, and image manipulation into a single hybrid control scheme.

Currently, we focused on the case where a single projector is used to achieve the target image. Since our prototype RSAR system (i.e., ETRI FRC [16]) is equipped with dual projectors, they can be collaboratively used to achieve the goal. For example, bigger target image can be projected using dual projectors. A setup of multiple projectors can be also used to make a projected image more precise and brighter, and to collaboratively enhance projectability on a more complex surface [3].

In this paper, we dealt with the geometric constraint of a target image (i.e., a convex quadrilateral). For a practical application, however, we need to consider various types of constraints and conditions such as illumination, visibility, resolution, and reflection on a projection surface [3].

We are interested in devising a unified, parallel scheme of hybrid control whereas our current method based on a serial collaboration among individual control schemes in kinematics, projection, and image manipulation.

Acknowledgement

This research has been supported by Korean Ministry of Knowledge Economy and Korea Research Council for Industrial Science & Technology with the Grant No. 2010-ZC1140, Development of Future Robotic Computer.

References

[1] R. Azuma, Y. Baillot, R. Behringer, S. Feiner, S. Julier, B. MacIntyre, “Recent advance in augmented reality”, IEEE Computer Graphics and Applications 21 (6), pp. 34-47, 2001.

[2] R. Hainich, The End of Hardware: Augmented Reality and Beyond, BookSurge, 2009.

[3] O. Bimber and R. Raskar, Spatial Augmented Reality: Merging Real and Virtual Worlds, A K Peters, Ltd., Wellesley, 2005.

[4] R. Raskar, A. Majumder, H.P.A. Lensch, O. Bimber, “Projectors for Graphics”, SIGGRAPH Course Notes, SIGGRAPH, LA, 2008.

[5] P. Mistry, and Maes, P., “SixthSense: a wearable gestural interface” ACM SIGGRAPH ASIA’09 Sketches, Article 11, 1, 2009.

[6] R. Raskar, Welch, G., Chen, W.-C., "Table-Top Spatially-Augmented Reality: Bringing Physical Models to Life with Projected Imagery," IEEE and ACM International Workshop on Augmented Reality, pp.64, 1999.

[7] R. Raskar, van Baar, J., Beardsley, P., Willwacher, T., Rao, S., Forlines, C., “iLamps: Geometrically Aware and Self-Configuring Projectors”. Proc. ACM SIGGRAPH’03, 2003.

[8] R. Raskar, Beardsley, P., van Baar, J., Wang, Y., Dietz, P., Lee, J., Leigh, D., Willwacher, T., “RFIG Lamps: Interacting with a Self-describing World via Photosensing Wireless Tags and Projectors”. Proc. ACM SIGGRAPH’04, 2004.

[9] N. Linder and Maes, P., “LuminAR: portable robotic augmented reality interface design and prototype”. Adjunct Proc. ACM UIST’10, 2010.

[10] J.-E. Lee, Miyashita, S., Azuma, K., Lee, J.-H., Park, G.-T., “Anamorphosis Projection by Ubiquitous Display in Intelligent Space”. Proc. Int. Conf. Universal Access in Human-Computer Interaction (UAHCI '09), pp. 209-217, 2009.

[11] R. Yang, Gotz, D., Hensley, J., Towles, H., Brown, M.S., “PixelFlex: a reconfigurable multi-projector display system”, Proc. Visualization’01, pp.167, 2001

[12] R. Ziola, Grampurohit, S., Landes, N., Fogarty, J., Harrison, B., "OASIS: Examining a Framework for Interacting with General-Purpose Object Recognition," Intel Labs Seattle Technical Report, 2010.

[13] K. Lai, Bo, L., Ren, X. and Fox D., "A Large-Scale Hierarchical Multi-View RGB-D Object Dataset," IEEE International Conference on Robotics and Automation, 2011.

[14] A. D. Wilson, Hrvoje Benko, “Combining multiple depth cameras and projectors for interactions on, above and between surfaces”, Proc. ACM UIST’10, October 03-06, 2010.

[15] B.R. Jones, Sodhi, R., Campbell, R.H., Garnett, G., Bailey, B.P., “Build Your World and Play In It: Interacting with Surface Particles on Complex Objects”, Proc. Int. Symp. Mixed and Augmented Reality’10, pp.165-174, 2010.

[16] H. Kim, Y.-H. Suh, J.-H. Lee, J. Cho, M. Lee, J. Yeom, B. Vladimirov, H.S. Kim and N.-S. Park, "Robotic Computer as a Mediator in Smart Environments," Toward Useful Services for Elderly and People with Disabilities (ICOST 2011), Lecture Notes in Computer Science, vol. 6719, pp.241-245, 2011. (Video Demo: http://goo.gl/oLGIU)

[17] J.-H. Lee, H. Kim, Y.-H. Suh, H. Kim, “Issues in control of a robotic spatial augmented reality system,” Transactions on CAD/CAM Engineers, 16(6), 2011. In Press.

[18] J.-H. Lee, “Inverse perspective projection of convex quadrilaterals,” Proc. Asian Conference on Design and Digital Engineering (ACDDE 2011, China), Aug., 2011.

[19] S.-Y. Park, and Park, G.G., “Active Calibration of Camera-Projector Systems based on Planar Homography”, Proc. Int. Conf. Pattern Recognition’10, pp. 320-323, 2010.

[20] R. Hartley, Multiple View Geometry in Computer Vision, 2nd Edition, Cambridge University Press, Cambridge, 2003.

Authorized licensed use limited to: University of Calgary. Downloaded on August 13,2021 at 18:36:22 UTC from IEEE Xplore. Restrictions apply.