# Accurate control of a pan-tilt system based on parameterization of rotational motion

JungHyun Byun, SeungHo Chae, and TackDon Han

Media System Lab, Department of Computer Science, Yonsei University, Republic of Korea

**Abstract**

*A pan-tilt camera system has been adopted by a variety of fields since it can cover a wide range of region compared to a single fixated camera setup. Yet many studies rely on factory-assembled and calibrated platforms and assume an ideal rotation where rotation axes are perfectly aligned with the optical axis of the local camera. However, in a user-created setup where a pan-tilting mechanism is arbitrarily assembled, the kinematic configurations may be inaccurate or unknown, violating ideal rotation. These discrepancies in the model with the real physics result in erroneous servo manipulation of the pan-tilting system.*

*In this paper, we propose an accurate control mechanism for arbitrarily-assembled pan-tilt camera systems. The proposed method formulates pan-tilt rotations as motion along great circle trajectories and calibrates its model parameters, such as positions and vectors of rotation axes, in 3D space. Then, one can accurately servo pan-tilt rotations with pose estimation from inverse kinematics of their transformation. The comparative experiment demonstrates out-performance of the proposed method, in terms of accurately localizing target points in world coordinates, after being rotated from their captured camera frames.*

**CCS Concepts**
•*Computing methodologies* → *Computer vision; Vision for robotics; Camera calibration; Active vision;*

## 1. Introduction

A pan-tilt platform consists of two motors, each rotating in *pan* and *tilt* directions. These 2 degrees of rotation freedoms grant the mounted system theoretically 360 degrees field-of-view. This is particularly beneficial to computer vision applications, for cameras can only obtain data from a field of view that is directed by the optical axis at a time [NLW*17]. To obtain scene information from a larger field of view, the camera had to be translated or rotated to capture a series of images. Various applications utilizes pan-tilt platforms from obvious video surveillance [DC03] to video conferencing, human-computer interaction, and augmented/mixed reality [WBIH12, BCYH17].

Despite continuous endeavors of the literature on the pan-tilting mechanics, many studies based their ground on fully factory-assembled and calibrated platforms such as in [DC03, WBIH12, LZHT15], which price from several hundreds to thousands of dollars. This may be problematic because the pan-tilting model may work seemingly error-free even if assuming an ideal rotation where rotation axes are aligned with the camera optical axis perfectly orthogonally, due to sound reliable assembly quality.

On the other hand, there is growing interest in user-created robots [Par11], where users create their own version of robots with such as consumer kits on the market [BCYH17]. In this case, it is unlikely that the exact kinematic specifications will be provided, for example in a CAD file format. Moreover, the mechanism may be fabricated and assembled in an do-it-yourself or arbitrary manner. All these factors attribute to erroneous, if even existent, kinematic equations, rendering kinematic control methods obsolete.

In this paper, we would like to address the issue of accurate pan-tilting manipulation even the pan-tilting kinematics are loosely coupled, or often unavailable. More specifically, we propose an operating mechanism of an arbitrarily assembled pan-tilt model with loose kinematics based on rotational motion modeling of the mounted camera. Our method is based on the pan-tilt model that is general enough to calibrate and compensate for assembly mismatches, such as skewed rotation axes or off-origin rotations.

## 2. Related Work

Calibrating pan-tilt motion of the camera has been broadly studied in the computer vision field, especially surveillance using pan-tilt-zoom (PTZ) camera. For example, Davis and Chen in [DC03] presented a method for calibrating pan-tilt cameras that incorporates a more accurate complete model of camera motion. Pan and tilt rotations were modeled as occurring around detached arbitrary axes in space, without the assumption of rotation axes aligned to camera optical axis. Wu and Radke in [WR13] introduced a camera calibration model for a pan-tilt-zoom camera, which explicitly reflects how focal length and lens distortion vary as a function of zoom scale. Using a nonlinear optimization, authors were able to accurately calibrate multiple parameters in one whole step. They also

investigated and analyzed multiple cases of pan-tilt errors to maintain the calibrated state even after extended continuous operations. Li et al. in [LZHT15] presented a novel method to online calibrate the rotation angles of a pan-tilt camera by using only one control point. By converting the non-linear pan-tilt camera model into a linear model according to sine and cosine of pan and tilt parameters, a closed-form solution could be derived by solving a quadratic equation of their tangents.

In the optics and measurement literatures, studies regarding calibration methods for a turntable or rotational axis have been proposed. Chen et al. in [CDCZ14] fixed a checkerboard plate on a turntable and captured multiple views of one pattern. Retrieved 3D corner points in a 360° view were used to form multiple circular trajectory planes, the equation and parameters of which were acquired using constrained global optimization method. Niu et al. in [NLW*17] proposed a method for calibrating the relative orientation of a camera fixed on a rotation axis, where the camera cannot directly 'see' the rotation axis. Utilizing two checkerboards, one for the rotating camera and one for the external observing camera, they were able to calibrate the relative orientation of the two cameras and rotation axis represented in the same coordinate system.

In this paper, we propose an operating mechanism of an arbitrarily assembled pan-tilt model with loose kinematics based on rotational motion modeling of the mounted camera. Our contributions can be summarized as follow:

- First, the proposed method models and calibrates the rotation of a generic pan-tilt platform by recovering directions and positions of its axes in 3D space, utilizing an RGB-D camera.
- Second, the proposed method is capable of manipulating servo rotations with respect to the camera, based on the inverse kinematic interpretation of its pan-tilt transformation model.

## 3. Pan-Tilt Rotation Modeling

Our goal is to model the rotational movement of a pan/tilting platform, so that we can estimate the pose of the RGB-D camera mounted on top of it. The platform rotates about arbitrarily assembled, independent axes [DC03] with loosely coupled kinematics. The structural model of such a setup is illustrated in Figure 1.



**(a)** *Pan-tilt assembly model.*     **(b)** *Rotation model formation.*
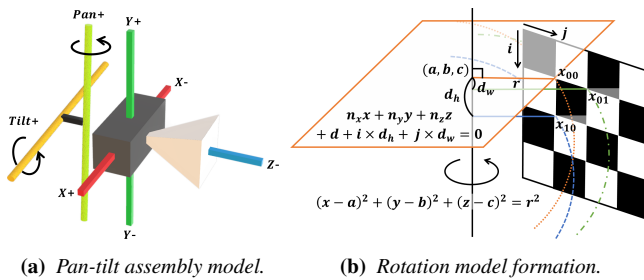
**Figure 1:** *Structural diagram of pan-tilt rotations with respect to the upper-left corner of the checkerboard.*

### 3.1. Rotation Parameters Acquisition

To model the movement of the motor rotation, we first calibrate parameters for the *pan* and *tilt* rotation. The calibration is a two-

step process, where we first estimate the direction vector of the rotation, and then estimate the center of the circular trajectory. The directions and centers of two rotations are all free variables.

When arbitrary points are rotated around a rotation axis, they create a closed circular rotation trajectory on a 3-dimensional plane, where the plane is perpendicular to the rotation axis and the circle center lies on the rotation axis. From the coordinates of a same point in rotated frames, the circular trajectory can be obtained using the least squares method [Sch06, DCYH13].

During the calibration, the camera captures multiple frames of a large checkerboard in front while it rotates, so that the checkerboard moves from one end of the field of view to another. Since the structure of the checkerboard is pre-known, all the rotation trajectories can be represented with respect to that of the top-leftmost corner. Then, we can parametrize the rotation with every corner of every frame and solve the objective function as a whole. If the checkerboard comprises $m$ corners in the vertical direction and $n$ corners in the horizontal direction and $l$ frames were taken throughout the calibration, we have total $l \times m \times n$ corners to globally optimize.

For the rotation of the upper-left corner, let us denote its rotation direction vector as $n = [n_x, n_y, n_z]^\mathsf{T}, ||n|| = 1$ and rotation circle center as $p = [a, b, c]^\mathsf{T}$. Then the rotation axis equation becomes

$$\frac{x-a}{n_x} = \frac{y-b}{n_y} = \frac{z-c}{n_z}, \quad (1)$$

and the rotation plane, which the upper-left corner is on, is $d$ away from the origin:

$$n_x x + n_y y + n_z z + d = 0. \quad (2)$$

Since all the rotation circles made from checkerboard corners are defined on the same rotation axis, the distance between their planes can be defined with respect to the indices of checkerboard corners. Let us denote the distances between the planes are $d_h$ and $d_w$ respectively in vertical and horizontal directions. Then for the corner at the $i$-th row and $j$-th column of the checkerboard, the rotation circle center becomes $p_{ij} = [a_{ij}, b_{ij}, c_{ij}]^\mathsf{T}$, where

$$
\begin{aligned}
a_{ij} &= a - n_x(i \times d_h + j \times d_w), \\
b_{ij} &= b - n_y(i \times d_h + j \times d_w), \\
c_{ij} &= c - n_z(i \times d_h + j \times d_w).
\end{aligned}
\quad (3)
$$

The ideal rotation trajectory will be modeled as a great circle, which is represented as the intersection of a plane and a sphere in 3D space. Here, the plane can be modeled as

$$n_x x + n_y y + n_z z + d + i \times d_h + j \times d_w = 0 \quad (4)$$

and the sphere, or the intersecting circle can be modeled as

$$(x - a_{ij})^2 + (y - b_{ij})^2 + (z - c_{ij})^2 = r_{ij}^2. \quad (5)$$

Let us denote the 3D vertex of the corner at the $i$th row and $j$th column of the $k$th captured checkerboard frame as $v_{ijk} = [x_{ijk}, y_{ijk}, z_{ijk}]^\mathsf{T}$. Then, we can setup the objective function where our goal is to find parameters $n_x, n_y, n_z, d, d_h, d_w$ that minimize the following error for the plane model:

$$\sum_{k=0}^{l-1} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} (n_x x_{ijk} + n_y y_{ijk} + n_z z_{ijk} + d + i \times d_h + j \times d_w)^2. \quad (6)$$

From Equation 3 and Equation 6, one can calculate parameters $a$, $b$, $c$, $r_{ij}$ that minimize the following error for the circle model:

$$\sum_{k=0}^{l-1}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}((x_{ijk}-a_{ij})^2+(y_{ijk}-b_{ij})^2+(z_{ijk}-c_{ij})^2-r_{ij}^2)^2. \quad (7)$$

The global least squares method is adopted to minimize errors of two objective functions with regard to entire coordinate variations of 70 checkerboard corners in all frames. This yields optimized parameters for the rotation axis calibration [CDCZ14].

### 3.2. Pan-Tilt Transformation Model

If we denote a 3D point taken in some *local* camera frame after rotating *tilt* and *pan* angles $P_{local}$, and the point before rotations $P_{world}$, the relationship can be written with the rotation model as

$$\begin{bmatrix} P_{world} \\ 1 \end{bmatrix} = T_{pan} R_{pan} T_{pan}^{-1} T_{tilt} R_{tilt} T_{tilt}^{-1} \begin{bmatrix} P_{local} \\ 1 \end{bmatrix}, \text{ where}$$

$$R(\theta) = \begin{bmatrix} C+n_x^2(1-C) & n_xn_y(1-C)-n_zS & n_xn_z(1-C)+n_yS & 0 \\ n_yn_x(1-C)+n_zS & C+n_y^2(1-C) & n_yn_z(1-C)-n_xS & 0 \\ n_zn_x(1-C)-n_yS & n_zn_y(1-C)+n_xS & C+n_z^2(1-C) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

$$C = \cos\theta, \ S = \sin\theta, \text{ and } T = \begin{bmatrix} 1 & 0 & 0 & a \\ 0 & 1 & 0 & b \\ 0 & 0 & 1 & c \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (8)$$

Here, $R_{tilt}$ is a 4×4 matrix that rotates around the direction vector $n = [n_x, \ n_y, \ n_z]^\mathsf{T}$ of the *tilt* axis and $T_{tilt}$ is a 4×4 matrix that translates by the coordinates of the pivot point $p = (a, \ b, \ c)$ of the *tilt* axis. Transformations regarding *pan* are analogous. Note that in the model *tilt* rotation comes before *pan* rotation. This is due to the kinematic configuration of the pan-tilt system we used. As shown in Figure 2, the tilting servo is installed on the panning arm. Thus to ensure unique rotation in *world* space, we *tilt* first, then *pan*.

### 3.3. Servo Control with Inverse Kinematics

Our scenario is that the users want to orient the camera so that the target object is located at the image center after the rotation. Specifically, the optical axis of the camera should pass through the target point. Then the task is rotating some point on the optical axis to a target point in *world* space. This task can be thought of as rotating the linear-actuated end-effector $P_{local} = [0, \ 0, \ sz]^\mathsf{T}$ (some point on the optical axis) of a robot arm (pan-tilting platform) to the target point $P_{dst} = [dx, \ dy, \ dz]^\mathsf{T}$, with unknown $\alpha$ *pan* and $\beta$ *tilt* angles. Using inverse kinematics notation on Equation 8, the problem becomes to find the parameter vector $\theta = [\alpha, \ \beta, \ sz]^\mathsf{T}$ of 2 rotations and 1 translation that minimizes the error $\|e\| = \|P_{dst}-P_{world}\|$. We adopt the Jacobian transpose method [Bus04] for Algorithm 1.

In our practice, we initialized $\alpha, \beta$ as 0 and $sz$ as $-\|P_{dst}\|$ and terminated the optimization if $\|e\| < 1$ *mm*. Maximum iteration count was set to 100, though 25 on average was enough to satisfy the condition. One pitfall here is that the magnitude difference can be extensive if the value of $sz$ is set in *mm*, for $\alpha$, $\beta$ range between $-1.57$ and $1.57$ radian while $sz$ may vary from 450 to 8000 *mm*. This difference causes too much fluctuations in $\alpha$, $\beta$ values, leading

---

**Algorithm 1** Inverse kinematics

    **input:** target point position $[dx, \ dy, \ dz]^\mathsf{T}$
    **output:** *pan* angle $\alpha$, *tilt* angle $\beta$, optical axis length $sz$
  **while** $P_{world}$ is too far from $P_{dst}$ **do**
      Calculate error vector $e = P_{dst} - P_{world}$.
      Setup parameter vector $\theta = [\alpha, \ \beta, \ sz]^\mathsf{T}$
      Compute Jacobian $J(e, \theta)$ of Equation 8 for the current pose
      Pick appropriate step $k = <e, JJ^\mathsf{T}e> / <JJ^\mathsf{T}e, JJ^\mathsf{T}e>$
      Calculate changes in parameters, $\triangle\theta = kJ^\mathsf{T}e$
      Apply changes to parameters $\theta \leftarrow \theta + \triangle\theta$
      Compute new $P_{world}$ position with forward kinematics 8

---

to far-from-optimal solutions. One solution is to optimize $sz$ in meter unit so that the domain difference to angles becomes minimum.

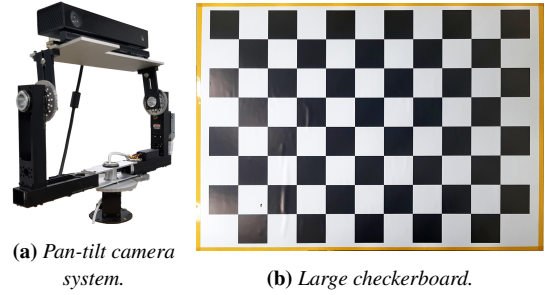### 4. Evaluation

### 4.1. System Configuration and Calibration



**(a)** *Pan-tilt camera system.*      **(b)** *Large checkerboard.*

**Figure 2:** *Pan-tilt experiment configuration.*

To validate the proposed setup, we set up an experimental setup that comprises an Microsoft Kinect v2 and two *pan* and *tilt* HS-785HB servos controlled with Arduino. For ease of explanation, we assume internal and external parameters of the color and depth cameras are already known. The checkerboard consists of 8 × 11 white-and-black checkers, each 100 *mm* × 100 *mm* in size.

We captured 28 frames by rotating the camera end-to-end in *pan* direction, and 11 frames in *tilt* rotation. With coordinates of 70 extracted 3D corners, we estimated parameter values in Equation 6 that form a linear function of multiple planes, using the least squares method. Then, we fitted corner trajectories to a circle with parameters in Equation 7. In Table 1, we show values of 6 key rotation parameters that govern pan-tilt transformation of Equation 8.

| | *Pan* rotation | *Tilt* rotation |
|---|---|---|
| $n_x$ | 0.011783038 | 0.998429941 |
| $n_y$ | 0.982956803 | -0.007633507 |
| $n_z$ | -0.183458670 | -0.055492186 |
| $a$ | -82.414993286 | -412.069976807 |
| $b$ | -458.764739990 | 153.644714355 |
| $c$ | 108.336227417 | 22.413515091 |

**Table 1:** *Rotation model parameter values in* mm.

## 4.2. Experiment Design

Here, we examine the performance of the proposed pan-tilt rotation model in terms of the accurate targeting capability. The system is tasked to adjust its attitude so that the target point is identical to the optical of center of the camera. We measure errors between the estimated pixels/vertices and actual captured pixels/vertices after the rotation. The coordinates of the 70 checkerboard corners when the system is at its rest pose, *i.e. pan=tilt=0*, are used as target points, constituting 70 trials in total.

We compare our result with the result produced using the Single Point Calibration Method (SPCM) of [LZHT15] as the baseline. To steer the system, we used inverse kinematics Algorithm 1 for the proposed method, while for SPCM *pan, tilt* values are calculated in closed form based on its geometrical model.
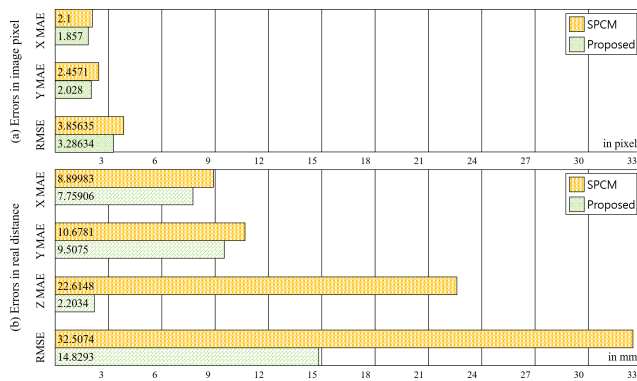
## 4.3. Results and Discussion



**Figure 3:** *Evaluation results of the proposed method and SPCM.*

Figure 3 summarizes evaluation results of two models. In [LZHT15], authors evaluated the model with only Root Mean Squared Errors (RMSE) of L2-norms in XY image pixels. Since we have depth values available, we also collected RMSEs of real distances (*mm*), and additionally measured Mean Absolute Errors (MAE) in each X, Y (and Z) directions in both pixels and *mm*s for further analysis.

In the graph, the proposed method outperforms SPCM in every metric, especially in Z directions in real distances. This can be explained by the omission of non-origin rotation centers (see Table 1) in SPCM. Also, the rotation axes are assumed to be identical to world XY axes which lead to additional errors. The error differences become small in image planes. We explain this is due to normalizing effects of perspective projections onto the image plane.

Beside the comparative analysis, the results show seemingly high error patterns. We conjecture a number of factors have attributed to this. First, there is the color-depth coordinate conversion. We detected checkerboard corners in color images, then converted them into depth image coordinates. Due to limitation in Kinect SDK, conversion in sub-pixel resolution was impossible, leading to error increases. Second, servo rotation manipulation could be inaccurate, maybe due to unreliable kinematics of low-grade servos, such as hysteresis, jitters or inaccurate pulse width-angle mapping.

## 5. Conclusion and Future Work

In this paper, we have proposed an accurate controlling method for arbitrarily-assembled pan-tilt camera systems based on the rotation transformation model and its inverse kinematics. The proposed model is capable of recovering rotation model parameters including positions and directions of axes of the pan-tilting platform in 3D space. The comparative experiment demonstrates outperformance of the proposed method, in terms of accurately localizing target world points in local camera frames captured after rotations.

In following future work, we would like to extend the proposed operating mechanism to a full-scale camera pose-estimation framework that can be used in projection-based augmented reality or point cloud registration and 3D reconstruction. We would also like to delve into hardware limitations discussed in 4.3 and improve the pan-tilt model to compensate errors further.

## 6. Acknowledgment

## References

[BCYH17] BYUN J., CHAE S., YANG Y., HAN T.: Air: Anywhere immersive reality with user-perspective projection. In *38th annual Eurographics conference* (2017), The Eurographics Association, pp. 5–8. 1

[Bus04] BUSS S. R.: Introduction to inverse kinematics with jacobian transpose, pseudoinverse and damped least squares methods. *IEEE Journal of Robotics and Automation 17*, 1-19 (2004), 16. 3

[CDCZ14] CHEN P., DAI M., CHEN K., ZHANG Z.: Rotation axis calibration of a turntable using constrained global optimization. *Optik-International Journal for Light and Electron Optics 125*, 17 (2014), 4831–4836. 2, 3

[DC03] DAVIS J., CHEN X.: Calibrating pan-tilt cameras in wide-area surveillance networks. In *In IEEE International Conference on Computer Vision* (2003), Citeseer. 1, 2

[DCYH13] DAI M., CHEN L., YANG F., HE X.: Calibration of revolution axis for 360 deg surface measurement. *Applied optics 52*, 22 (2013), 5440–5448. 2

[LZHT15] LI Y., ZHANG J., HU W., TIAN J.: Method for pan-tilt camera calibration using single control point. *JOSA A 32*, 1 (2015), 156–163. 1, 2, 4

[NLW*17] NIU Z., LIU K., WANG Y., HUANG S., DENG X., ZHANG Z.: Calibration method for the relative orientation between the rotation axis and a camera using constrained global optimization. *Measurement Science and Technology 28*, 5 (2017), 055001. 1, 2

[Par11] PARK W.: Philosophy and strategy of minimalism-based user created robots (ucrs) for educational robotics-education, technology and business viewpoint. *International Journal of Robots, Education and Art 1*, 1 (2011), 26–38. 1

[Sch06] SCHAFFRIN B.: A note on constrained total least-squares estimation. *Linear algebra and its applications 417*, 1 (2006), 245–258. 2

[WBIH12] WILSON A., BENKO H., IZADI S., HILLIGES O.: Steerable augmented reality with the beamatron. In *Proceedings of the 25th annual ACM symposium on User interface software and technology* (2012), ACM, pp. 413–422. 1

[WR13] WU Z., RADKE R. J.: Keeping a pan-tilt-zoom camera calibrated. *IEEE transactions on pattern analysis and machine intelligence 35*, 8 (2013), 1994–2007. 1