International Journal on Advanced Science Engineering Information Technology

Vol.12 (2022) No. 4
ISSN: 2088-5334

Karate Video Reconstruction to Observe Player Motion and Flow of Competition

Kazumoto Tanaka [a,*]

[a] *Faculty of Engineering, Kindai University, 1, Umenobe, Takaya, Higashi-Hiroshima, 739-2116, Japan*
Corresponding author: [*]kazumoto@hiro.kindai.ac.jp

***Abstract*— It is well known that sports skill learning for trainees is facilitated via video observation of the players' actions and the overall flow of competition on the playing field. A video zoomed in on a player's actions is suitable for learning the player's technical skills, and a video showing all the players on the competition court is suitable for learning tactical skills such as player formation. This study aims to establish a method that can make narrow field-of-view videos available for observing the player positioning throughout the competition court. This paper focuses on karate competition videos and proposes an image processing method that extracts the image of the player area from the video frame and superimposes it on a standard karate court model image. Before the superimposition, the court model image is homographically transformed to fit the court in the video image. Using this approach, a player-focused video is reconstructed as a video of the entire competition field. For the reconstruction, a vertex matching method is developed for homography matrix calculation, and another calculation method for homography matrix calculation using only two straight lines is also developed for images that lack vertex information. Finally, an experimental result shows that a karate player-focused video is transformed successfully into a video of the entire karate competition field. Future work will focus on experimentally verifying the effectiveness of the method for skill learning.**

***Keywords*—Video reconstruction; video observation; homographic transformation; camera calibration.**

Manuscript received 10 Mar. 2021; revised 6 May 2021; accepted 22 Jun. 2021. Date of publication 31 Aug. 2022.

I. INTRODUCTION

It is well known that a Sport's technical/tactical skill development is facilitated by video observation of the players' actions and the overall flow of competition on the playing field [1], [2]. However, when the video camera zooms in on a player to observe her/his action, it may become difficult to see the entire flow of the competition. To support narrow field-of-view video observation, this study develops a method that extracts the image of the player area from a zoomed-in video frame and superimposes this area on a model image of the entire Sports court. This way, a video of part of the real court can be converted into a video of the entire court, making it easier to observe the players' actions and the overall flow of competition.

Before the superimposition, the court model image must be homographically transformed to fit the court in the video image. Numerous studies have used homography matrix calculations for Sports analysis [3]–[18]. Most of them have focused on Sports such as soccer and basketball; the competition court is assumed to be a plane, and the homography matrix between the court model and the court in the video images is calculated using a white line drawn on the court. However, there are two main issues on the homography matrix estimation that have been tackled by recent studies. The one is the estimation accuracy [5]–[8], and the other is the estimation with insufficient geometric information of the court [9]–[18]. The proposed method mainly addressed the latter problem.

When the video images lack geometric information about the court due to a narrow field-of-view or occlusion, it is difficult to match the white lines (or intersections of white lines) seen by the video camera on the playing field with a standard court model. Thus, the homography matrix estimation may fail. Since the homography matrix calculation is almost equivalent to camera calibration, it is important to look at studies on camera calibration in Sports images. One promising technique reduced the number of white lines required for camera calibration by targeting the stadium broadcast camera. This camera can be treated as a fixed-position pan-tilt-zoom (PTZ) camera, and consequently, the number of camera parameters to be estimated is reduced [9].

Using machine learning to tackle the lack of feature points, Chen *et al.* [10] estimated the camera pose using a random forest. The approach learned the relationship between the image coordinates of the feature points of a soccer court image recorded using a broadcast video camera and the 3D parameters of the feature points expressed in the camera coordinate system. As a result, calibration was possible even with only two feature points. Sharma *et al.* [11] were able to calibrate images with no feature points by using a large dictionary that collected pairs of court edge images and homography matrices of various camera views. Chen and Little [12] labeled each pair of court edge images as similar or dissimilar in terms of the pose of the camera that captured them and trained a Siamese network to learn the edge image pairs (as input) and their labels (as output). As a result, the network could estimate the camera pose for a target edge image. However, all of the abovementioned studies pertained to PTZ cameras and did not enable calibration for arbitrary camera poses.

Other approaches employ the iterative closest point (ICP) algorithm [13], image stitching [14], [15], Markov Random Field (MRF) [16], and Deep Neural Network (DNN) [17], [18] have also been proposed. Lu *et al.* [13] proposed a method to calibrate the keyframes of a basketball video and applied the ICP algorithm to match these key frames and the remaining frames to calibrate the remaining frames using the court's edges. The method solved the problem of the lack of edge points due to occlusion by using the stable feature points of markers such as logos drawn on the court for point matching. However, no logos are present on martial arts courts, such as those of karate, which is the subject of this study.

Wen *et al.* [14] utilized image stitching to generate a panoramic image from multiple frames of a basketball video. It was expected that a panorama would have sufficient information for finding correspondences between the image and the court model. The homographic transformation between each frame composing the panoramic image and the court model could be calculated from the panorama. Frames not showing the entire court could be calibrated via the panoramic image by calculating a homographic map to the calibrated images. However, image stitching requires the correspondence of four points between the frames, and if the number of correspondences is insufficient, image stitching cannot be performed. In the current study, we developed a method to calculate a homography matrix when no corresponding points are present, only two straight lines.

Homayounfar *et al.* [16] employed a branch and bound inference in a MRF to detect four sidelines of the playing field to estimate the homography matrix. As a result, the method reduces the number of degrees of freedom of the homography from 8 to 4. First, they detect two vanishing points of the field by using a deep semantic segmentation network, and the vanishing points are used to constrain the search space for the inference. However, the estimation accuracy depends on two vanishing points. It can be said that there is a problem with robustness. Our method also uses vanishing points but for a different purpose.

Citraro *et al.* [17] proposed a DNN that detects keypoints to compute the homography matrix. The key points consist of the intersections of white lines and the locations of players, and the player locations are used to refine the homography. Meanwhile, Nie *et al.* [18] employed a grid of uniformly distributed keypoints over the entire field instead of line intersections and proposed a DNN that detects the keypoints. The feature of the keypoint is defined as a normalized distance map to its nearest line and marking (e.g., yard-line number) on the field. The proposed method showed a robust homography estimation using the keypoints. However, applying these methods to karate video images is difficult, as karate players are often filmed with a narrow field-of-view. The player area in the image will interfere with the keypoint detection.

This study is focused on martial arts such as karate, which have not been considered in previous studies. In this domain, there is more interest in the player's actions and eye movements than in Sports such as soccer and basketball; it is, therefore, necessary to film competitions with a narrow field-of-view camera focused on the players (Fig. 1(a). Furthermore, because a player's movement out of the court causes a penalty, the positional relationship between the players is also important for observational learning. Thus, the study aims to develop a method in which player-focused video is reconstructed as a video of the entire competition field by superimposing the player area on a homographically transformed karate court model image (Fig. 1).

In addition, the authors address the case in which an amateur karate-practitioner films competitions with a portable video camera or smartphone camera and uses this video for skill learning. Many amateur practitioners employ such techniques; as they film the competition from the seats in the auditorium (Fig. 2) with handheld cameras, the method developed for the PTZ camera cannot be applied.

The video reconstruction consists of three main techniques: homography transformation, player extraction, and superimposition. Recent computer vision studies have proposed powerful human detection and segmentation frameworks based on deep learning, e.g., Mask R-CNN [13]. This framework can be applied to player extraction. The superimposition can be done easily if the homographically transformed court model image overlaps the court in the video image.

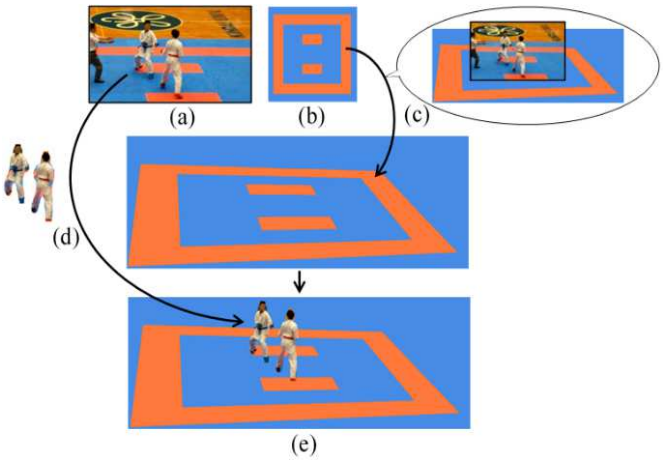

Fig. 1 Overview of the video reconstruction: (a) video image; (b) court model image; (c) the court model image is homographically transformed so that the image fits the court in the video image; (d) player extraction and superimposition; (e) reconstructed video image.

Fig. 2 Karate tournament viewed from the auditorium seating.

II. Materials and Method

A. Overview

This paper proposes two methods of image processing for the homography matrix calculation. One method is related to the correspondence between the video image and the court model. As shown in Fig. 1, a mat divided into two colored figures is used as the competition court of karate, and the vertices of these figures are used as the feature points. Many previous studies on ball games detected the vertex by calculating the intersection of white lines in the image for homography matrix calculation. However, in martial arts, the edge extracted from an image may be short owing to the players' narrow camera field of view and occlusion. Consequently, the accuracy of the direction of the straight line estimated from the edge becomes insufficient, leading to issues in the calculation accuracy of the vertex coordinates. Some corner detection methods using the Harris corner detector and other techniques have been proposed for vertex detection; however, they encounter difficulties in situations involving considerable noise. Template matching using a small square area surrounding a vertex as a template can help detect and identify vertices by using multiple templates according to the vertex type; however, this approach is not robust to image rotation or deformation. Thus, in this study, perspective distortion correction [14] is applied to transform the image to ensure that the edges of the colored figures are horizontal/vertical; subsequently, vertex detection is performed by template matching. Furthermore, by using the labels indicating the vertex types, the vertices in the image and the vertices of the court model are each represented by label sequences and associated by label sequence matching. This processing method is described in detail in sub-section B.

In general, a homography matrix calculation requires at least four points. A second method proposed in this study addresses the situation where the four vertices are not in the image, owing to a narrow camera field-of-view and occlusion. Assuming that the homography matrices can be calculated for other images close to the image, we develop a method that enables homography matrix calculation if only two straight lines exist in the image. This is described in sub-section C.

B. Homography Matrix Calculation by Vertex Matching

The homography matrix between each video frame and the court model is calculated for transforming the court model. The court model is virtually placed to match the actual competition court position. To match the feature points required for the matrix calculation, template matching is performed using four or more vertices of the color figures of the court as the feature points. Figure 3 shows the general flow up to matching. First, the mat area of the image is extracted using the two colors of the mat and is binarized, with the two colors being black and white (see Fig. 3(c)). The vertices of the white blobs are the feature points. However, the vertices that occur when the player or referee overlaps with the color figures (yellow circles in Fig. 3(b)) are not used for vertex matching. The players and referees are extracted by human region extraction (Mask R-CNN was used in the study: see Fig. 3(d)).

Eight types of vertices are present in the court model (Fig. 3(f)), as shown in Fig. 4, and are used as templates. However, image rotation or deformation can lead to template matching failure. Therefore, the following image correction is performed as preprocessing for the template matching:

The color figures of the court have two types of mutually perpendicular edges. Because it is impossible to film the court from directly overhead, distortion due to perspective always occurs, and the straight lines of these edges on the image meet at one of the two vanishing points. This perspective distortion is corrected by the homographic transformation using the vanishing points [14] (see Fig. 5).

The vanishing points are obtained by applying the method of Wen *et al*. [14]. First, for each blob, the straight lines (ρ_i,θ_i) are detected by Hough transform for the contours of the blobs. Next, all the straight lines are clustered by θ using the k-means (k = 2) method. For these two clusters, the result of ST-transformation (ρ_i,θ_i) of each cluster lies on a straight line in the ST space [14], and the parameters of the straight line are obtained by removing the outliers by RANSAC represent the vanishing points. However, those straight lines in the image with edge pixels that occur owing to the overlap of the players or referee with the color figures are removed.

The sides of the color figures of the court become horizontal/vertical when a homographic transformation based on the vanishing point is performed (see Fig. 3(e)). Subsequently, the vertices of the corrected court image and the vertices of the court model are matched. This image correction makes the template matching robust and facilitates vertex registration. In other words, if, for example, the vertices in Fig. 4 are labeled from the left with an integer label starting from 1, the vertex arrangement of the court model can be represented by the two-dimensional (2D) character string shown in Fig. 6. The detected vertices in the image can also be represented in the same manner (for example, in Fig. 7), and the vertex matching can be executed as an elastic string matching. Consequently, the homography matrix between the court model and the court in the video image can be calculated based on the vertex matching result. However, as a result of this matching, multiple types of correspondence may be obtained. In this case, the image is homographically transformed based on each correspondence. Subsequently, the correspondence that maximizes the degree of coincidence between the color figures of the transformed image and the color figures of the court model is selected. The implemented vertex matching procedure is presented in Appendix A.

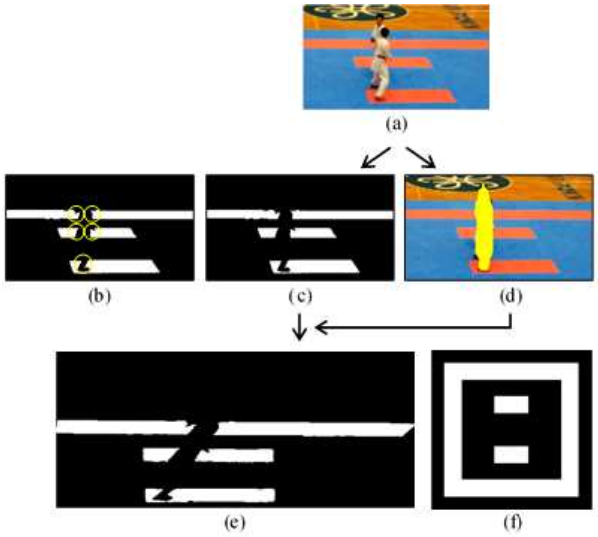

Fig. 3 Overview of the vertex matching preprocessing.

Fig. 4 Vertex detection template.

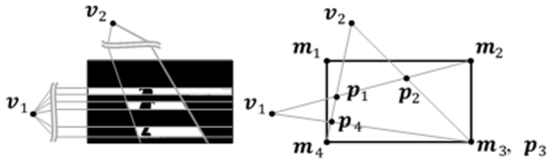

Fig. 5 Perspective distortion correction using vanishing points. Left: two vanishing points $\boldsymbol{v}_1, \boldsymbol{v}_2$ from the straight lines extracted by Hough transform. Right: $\boldsymbol{m}_1$ - $\boldsymbol{m}_4$ are the four corner points of the image frame, and $\boldsymbol{p}_1$ - $\boldsymbol{p}_4$ are the intersection points between $\overline{\boldsymbol{v}_1\boldsymbol{m}_2}$ and $\overline{\boldsymbol{v}_2\boldsymbol{m}_4}$, $\overline{\boldsymbol{v}_1\boldsymbol{m}_2}$ and $\overline{\boldsymbol{v}_2\boldsymbol{m}_3}$, $\overline{\boldsymbol{v}_1\boldsymbol{m}_3}$ and $\overline{\boldsymbol{v}_2\boldsymbol{m}_3}$, $\overline{\boldsymbol{v}_1\boldsymbol{m}_3}$ and $\overline{\boldsymbol{v}_2\boldsymbol{m}_4}$, respectively. The perspective distortion can be corrected through a homography transformation that matches $\boldsymbol{p}_1$ to $\boldsymbol{m}_1$, $\boldsymbol{p}_2$ to $\boldsymbol{m}_2$, $\boldsymbol{p}_3$ to $\boldsymbol{m}_3$, and $\boldsymbol{p}_4$ to $\boldsymbol{m}_4$ [14]. Since there are other arrangements of vanishing points, an appropriate set of point matchings is employed corresponding to the arrangement. The vanishing point is a point at infinity if the parallel lines are horizontal or vertical. If $\boldsymbol{v}_1$ in the figure is a point at infinity $\overline{\boldsymbol{v}_1\boldsymbol{m}_2}$ overlaps $\overline{\boldsymbol{m}_1\boldsymbol{m}_2}$ and $\overline{\boldsymbol{v}_1\boldsymbol{m}_3}$ overlaps $\overline{\boldsymbol{m}_4\boldsymbol{m}_3}$.

1	0	0	0	0	2
0	5	0	0	6	0
0	0	1	2	0	0
0	0	3	4	0	0
0	0	1	2	0	0
0	0	3	4	0	0
0	7	0	0	8	0
3	0	0	0	0	4

Fig. 6 Two-dimensional label array representing the court model.

1	2
3	4
1	2
3	4

Fig. 7 Left: A binarized image with perspective distortion correction (yellow circles are added on vertices). Right: 2D label array representing the arrangement of vertices in the binarized image.

C. Homography Matrix Calculation using Two Straight Lines

The method described in the previous section is valid when the following conditions hold:

Condition 1: Two clusters of two or more straight lines parallel to each other in the image, where the straight lines are determined for the sides in the color figures.
Condition 2: A rectangle can be created with the vertices that have been matched.

If an image that does not satisfy these conditions contains at least two straight lines, and another image that includes the same straight lines has a homography matrix that has been calculated, the matrix for the problem image can be calculated using the technique below. It is assumed that the camera position is fixed, and only the camera direction (camera pose) changes with the displacement of the players. By observing people filming from the auditorium, the author found that they tracked the individual players by performing wrist rotation with the handheld video camera. Therefore, the assumption can be considered reasonable.

Let *image_i′* be an image that does not satisfy Conditions 1 and 2. First, the method finds an *image_i* that matches *image_i′* for template matching using the circumscribed quadrangular region, including all blobs of the binarized *image_i′* as a template. The search is executed beginning with the image close to *image_i′* at the time and the image for which the transformation matrix has been obtained. Next, the straight-line detection and clustering described in the previous section are performed for the images, and a straight line of *image_i* and a straight line of *image_i′* are associated by a nearest neighbor search in the Hough space.

Let O_c-$\mathrm{X}_c\mathrm{Y}_c\mathrm{Z}_c$ and $\mathrm{O}_{c'}$-$\mathrm{X}_{c'}\mathrm{Y}_{c'}\mathrm{Z}_{c'}$ be the camera coordinate systems when *image_i* and *image_i′* are captured, respectively. By the assumption that the camera position is fixed, $\mathrm{O}_c = \mathrm{O}_{c'}$. We express the straight lines on *image_i* and *image_i′* as $\mathrm{a}x + \mathrm{b}y + \mathrm{c} = 0$ and $\mathrm{a}'x' + \mathrm{b}'y' + \mathrm{c}' = 0$, using the 2D coordinate systems of *image_i* and *image_i′*, respectively. Thus, the expression for the normal vector of the plane, including the straight line in *image_i* and the origin in the camera coordinate system O_c-$\mathrm{X}_c\mathrm{Y}_c\mathrm{Z}_c$ is $({}^{c}(\mathrm{a},\mathrm{b},\mathrm{c}/f)^{\mathrm{T}})$. Similarly, the normal vector of the plane, including the straight line in *image_i′* and the origin in the camera coordinate system $\mathrm{O}_{c'}$-$\mathrm{X}_{c'}\mathrm{Y}_{c'}\mathrm{Z}_{c'}$ is $N\left({}^{c'}(\mathrm{a}',\mathrm{b}',\mathrm{c}'/f)^{\mathrm{T}}\right)$ (see Fig. 8). Here, $N(\cdot)$ indicates the normalization into a unit vector. Furthermore, we define the world coordinate system as O_w-$\mathrm{X}_w\mathrm{Y}_w\mathrm{Z}_w$ at the real court plane. Let ${}^{w}_{c}\boldsymbol{R}$ and ${}^{w}_{c'}\boldsymbol{R}$ be the coordinate transformation matrices from O_c-$\mathrm{X}_c\mathrm{Y}_c\mathrm{Z}_c$ to O_w-$\mathrm{X}_w\mathrm{Y}_w\mathrm{Z}_w$ and $\mathrm{O}_{c'}$-$\mathrm{X}_{c'}\mathrm{Y}_{c'}\mathrm{Z}_{c'}$ to O_w-$\mathrm{X}_w\mathrm{Y}_w\mathrm{Z}_w$, respectively. Because the two normal vectors expressed in the world coordinate system are the same,

$$ {}^{w}_{c'}\boldsymbol{R}N((\mathrm{a}',\mathrm{b}',\mathrm{c}'/f)^{\mathrm{T}}) - {}^{w}_{c}\boldsymbol{R}N((\mathrm{a},\mathrm{b},\mathrm{c}/f)^{\mathrm{T}}) = \boldsymbol{0}, \quad (1) $$

The previously calculated homography matrix $\boldsymbol{H}$ from the court model to *image_i* is

$$ \boldsymbol{H} = \begin{pmatrix} f & 0 & c_x \\ 0 & f & c_y \\ 0 & 0 & 1 \end{pmatrix} ({}^{c}_{w}\boldsymbol{R}.col(1) \quad {}^{c}_{w}\boldsymbol{R}.col(2) \quad {}^{c}\boldsymbol{t}_{c\to w}) \quad (2) $$

where ${}^{c}_{w}\boldsymbol{R} = {}^{w}_{c}\boldsymbol{R}^{\mathrm{T}}$, *col*(*i*) is the *i*-th column of the matrix, ${}^{c}\boldsymbol{t}_{c\to w}$ denotes the $\overrightarrow{\mathrm{O}_c\mathrm{O}_w}$ expressed by O_c-$\mathrm{X}_c\mathrm{Y}_c\mathrm{Z}_c$ and is calculated as

$$\left(\begin{pmatrix} f & 0 & c_x \\ 0 & f & c_y \\ 0 & 0 & 1 \end{pmatrix}^{-1} \boldsymbol{H}\right).col(3) \quad (3)$$

(c_x, c_y) is a principal point calibrated in advance, and f is the focal length calculated using the world coordinates of the vertices of the color figures in the video images.
${}^{c}_{w}\boldsymbol{R}.column(2)$ is calculated as

$$ {}^{c}_{w}\boldsymbol{R}.col(3) = \left(\begin{pmatrix} f & 0 & c_x \\ 0 & f & c_y \\ 0 & 0 & 1 \end{pmatrix}^{-1} \boldsymbol{H}\right).col(1) \times \left(\begin{pmatrix} f & 0 & c_x \\ 0 & f & c_y \\ 0 & 0 & 1 \end{pmatrix}^{-1} \boldsymbol{H}\right).col(2), \quad (4)$$

${}^{w}_{c}\boldsymbol{R}$ is obtained using Eqs. (2) and (4).

For another pair of straight lines associated with each other in the two images,

$$ {}^{w}_{c'}\boldsymbol{R}N((\mathrm{g}', \mathrm{h}', \mathrm{i}'/f)^{\mathrm{T}}) - {}^{w}_{c}\boldsymbol{R}N((\mathrm{g}, \mathrm{h}, \mathrm{i}/f)^{\mathrm{T}}) = \boldsymbol{0}, \quad (5)$$

is also obtained in the same manner. ${}^{w}_{c'}\boldsymbol{R}$ is determined using Eqs. (1) and (5). In other words, the Euler angles $\boldsymbol{\theta} \equiv (\theta_x, \theta_y, \theta_z)^{\mathrm{T}}$ of ${}^{w}_{c'}\boldsymbol{R}$ are obtained using Gauss-Newton method. Consequently, the homography matrix $\boldsymbol{H}'$ for *image_i'* is

$$\boldsymbol{H}' = \begin{pmatrix} f & 0 & c_x \\ 0 & f & c_y \\ 0 & 0 & 1 \end{pmatrix} \cdot \left({}^{c'}_{w}\boldsymbol{R}.col(1) \quad {}^{c'}_{w}\boldsymbol{R}.col(2) \quad {}^{c'}_{w}\boldsymbol{R}\,{}^{w}_{c}\boldsymbol{R}\,{}^{c}\boldsymbol{t}_{c\to w}\right) \quad (6)$$

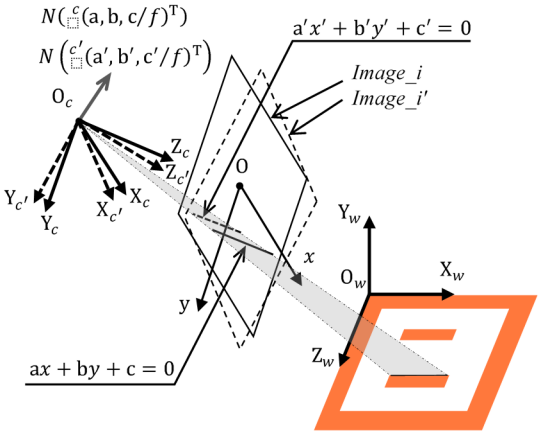

Fig. 8 Relationship among the camera coordinate systems; normal vector of the plane that includes the origin of the camera coordinate system; and the straight lines on the court, in the image, when the camera origin is fixed.

III. Results and Discussion

As an experiment to validate the proposed method, video reconstruction using a video of a karate competition was performed. The video was filmed using a smartphone camera (1280 x 720). The reconstruction was performed on a PC (Windows OS, 2.7 GHz Intel Core i5 CPU). The reconstruction (except for player extraction for each image) required, on average, approximately 300 msec. In this study, robust template matching for homography matrix calculation has been realized by correcting the distortion of the image due to perspective. The implemented elastic matching eliminates the incorrect vertex correspondence at the end; however, it would be possible to improve the efficiency through pruning in the middle of the matching process, and as a result, the calculation time can be expected to be shortened.

The result is shown in Fig. 9. Fig. 9(d) does not satisfy Conditions 1 or 2, owing to the occlusion by the players. Thus, the matrix for the image was calculated using the proposed method based on two straight lines. The two orthogonal sides of a color figure of the karate court were selected for this calculation.

The method for calculating the homography matrix with two straight lines would be the use of two arbitrary straight lines (unless the two lines are parallel and the relationship between O_c-$X_cY_cZ_c$ and $O_{c'}$-$X_{c'}Y_{c'}Z_{c'}$ is the rotation of the optical axis in the direction of the lines). However, in practice, the calculation accuracy seems to be better when the two orthogonal straight lines are selected; therefore, this technique has been employed in the experiment. If the number of straight lines is less than two, the method cannot be used to calculate the homography. However, if the camera's field of view is enough for both players to be in view, such a problem hardly ever occurs in the real world. However, even if the problem does occur, the image (*image_i'*) can be used without any processing as a part of reconstructed video images.

We used the intersection over union (IoU) score to evaluate the accuracy of the reconstruction. Thus, the input image (image of karate competition) was transformed using the inverse matrix of the homography matrix H, which is the transformation matrix from the court model to the input image (see Fig. 10). Using this transformed image, IoU was calculated by Eq. (7).

$$\mathrm{IoU} = \frac{(\boldsymbol{H}^{-1}\boldsymbol{F}_i)\mathrm{AND}(\boldsymbol{F}_m)}{(\boldsymbol{H}^{-1}\boldsymbol{F}_i)\mathrm{OR}(\boldsymbol{F}_m)}, \quad (7)$$

where $\boldsymbol{F}_i$ and $\boldsymbol{F}_m$ denote the colored figure of the karate competition court of the input image and the figure of the court model, respectively. However, the occlusion area of the figure of the input image by players was manually repaired. The average of IoU calculated by 17 images from different competition scenes was 0.924. Most of the related studies are on soccer and basketball competition scenes, and the latest study for improving transformation accuracy showed 0.975 [8]. Since the size and shape of the competition courts are different, it is not possible to simply compare our results with them, but it is difficult to obtain the information necessary for homography transformation because the proportion of the competition area occupied by karate players is larger than that of ball games such as soccer. With that in mind, the result of our study is at a high level.

Next, a Full HD movie created from the reconstruction was shown to a university karate club coach. He gave his impression as a simple evaluation: he was satisfied because the players' positioning and detailed actions could be observed simultaneously, and nothing extra was present in the reconstructed images.

In addition, the proposed method has the potential to be expanded to other martial arts. The method can be applied without any modification for competitions that use a court similar to karate (e.g., judo). It can be said that this study has contributed to expanding the scope of research on homography matrix calculation for Sports analysis, from ball games to martial arts.

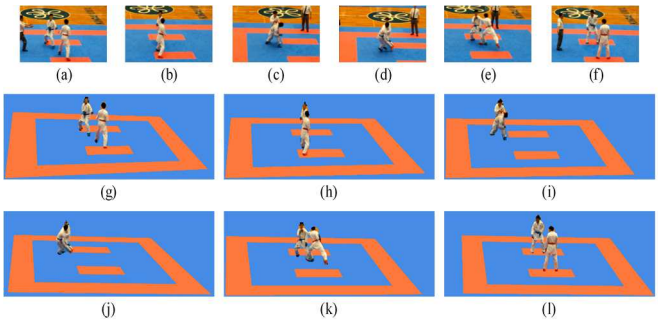

Fig 9. Original video images ((a)-(f)). Reconstructed images ((g)-(l)).

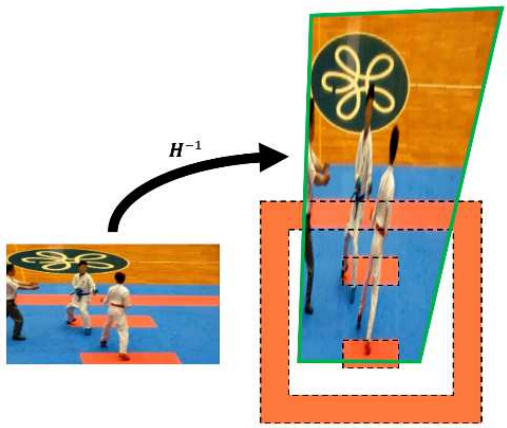

Fig 10. Karate competition image (left) was transformed by inverse homography matrix H^{-1} for IoU calculation. The red figures in the visible area surrounded by the green line are the target area of IoU (right).

IV. Conclusion

A novel video reconstruction method to observe players' motion and the flow of karate competition has been proposed in this paper. The method transforms the karate court model image to fit the court in the video image and superimposes images of the players extracted from the video image onto this transformed court model image. A vertex matching method using perspective distortion correction for homography matrix calculation and another homography matrix calculation method using only two straight lines for images that lack vertex information were developed in this study. As the result of the experiment using images extracted from a karate competition video, the image transformation accuracy of the method was approximately 92%. Since the latest studies have targeted soccer images, it is difficult to obtain the information necessary for homography transformation because the proportion of the competition area occupied by karate players is larger than that of ball games such as soccer. It can be said that the result of our study is at a high level. Through this study, we could extend the research of homography matrix calculation for Sports analysis for ball games to martial arts. Further work will focus on experimentally verifying the effectiveness of the video observation method.

Acknowledgment

This work was supported by JSPS KAKENHI Grant Number 18K02876.

References

[1] F. Potdevin, O. Vors, A. Huchez, M. Lamour, K. Davids, and C. Schnitzler, "How can video feedback be used in physical education to support novice learning in gymnastics? Effects on motor learning, self-assessment and motivation," *Physical Education and Sport Pedagogy*, vol. 23, no. 6, pp. 559-574, Jun. 2018.

[2] S. B. Nicholls, N. James, E. Bryant, and J. Wells, "The implementation of performance analysis and feedback within Olympic sport: The performance analyst's perspective,". *International Journal of Sports Science & Coaching*, vol. 14, no.1, pp. 63-71, Oct. 2018.

[3] D. Farin, J. Han, and P. H. N. de With, "Fast Camera Calibration for the Analysis of Sport Sequences," in *Proc. IEEE International Conference on Multimedia and Expo*, Amsterdam, Netherlands, 2005, DOI: 10.1109/ICME.2005.1521465.

[4] J. Han, D. Farin and P. H. N. de With, "Broadcast Court-Net Sports Video Analysis Using Fast 3-D Camera Modeling, " *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, no. 11, pp. 1628-1638, Nov. 2008, DOI: 10.1109/TCSVT.2008.2005611.

[5] Q. Yao, A. Kubota, K. Kawakita, K. Nonaka, H. Sankoh, and S. Naito, "Fast Camera Self-Calibration for Synthesizing Free Viewpoint Soccer Video," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, New Orleans, LA, 2017, pp. 1612-1616, DOI: 10.1109/ICASSP.2017.7952429.

[6] S. Liu, J. Chen, C. Chang and Y. Ai, "A New Accurate and Fast Homography Computation Algorithm for Sports and Traffic Video Analysis, " *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 28, no. 10, pp. 2993-3006, Oct. 2018, DOI: 10.1109/TCSVT.2017.2731781.

[7] W. Jiang, J. C. Gamboa Higuera, B. Angles, W. Sun, M. Javan, and K. M. Yi, "Optimizing Through Learned Errors for Accurate Sports Field Registration," in *Proc*. IEEE Winter Conference on Applications of Computer Vision, Snowmass, CO, 2020, pp. 201-210, DOI: 10.1109/WACV45572.2020.9093581.

[8] H. Tsurusaki, K. Nonaka, R. Watanabe, T. Konno, and S. Naito, "Sports Camera Calibration Using Flexible Intersection Selection and Refinement," *ITE Transactions on Media Technology and Applications*, vol. 9, no. 1, pp.95-104, Jan. 2021, DOI: https://doi.org/10.3169/mta.9.95.

[9] G. Thomas, "Real-time Camera Tracking Using Sports Pitch Markings," *Journal of Real-Time Image Processing*, vol. 2, pp. 117-132, 2007, DOI: 10.1007/s11554-007-0041-1.

[10] J. Chen, F. Zhu, and J. J. Little, "A Two-Point Method for PTZ Camera Calibration in Sports," in *Proc. IEEE Winter Conference on Applications of Computer Vision*, Lake Tahoe, NV, 2018, pp. 287-295, DOI: 10.1109/WACV.2018.00038.

[11] R. A. Sharma, B. Bhat, V. Gandhi, and C. V. Jawahar, "Automated Top View Registration of Broadcast Football Videos," in *Proc. IEEE Winter Conference on Applications of Computer Vision*, Lake Tahoe, NV, 2018, pp. 305-313, DOI: 10.1109/WACV.2018.00040.

[12] J. Chen, and J. J. Little, "Sports Camera Calibration via Synthetic Data," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, Long Beach, CA, 2019.

[13] W.-L. Lu, J.-A. Ting, J. J. Little, and K. P. Murphy, "Learning to Track and Identify Players from Broadcast Sports Videos," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 35. no. 7, pp. 1704-1716, Jul. 2013, DOI: 10.1109/TPAMI.2012.242.

[14] P.-C. Wen, W.-C. Cheng, Y.-S. Wang, H.-K. Chu, N. C. Tang, and H.-Y. M. Liao, "Court Reconstruction for Camera Calibration in Broadcast Basketball Videos," IEEE Transactions on Visualization and Computer Graphics, vol. 22, no. 5, pp. 1517-1526, May 2016, DOI: 10.1109/TVCG.2015.2440236.

[15] R. Zeng, R. Lakemond, S. Denman, S. Sridharan, C. Fookes, and S. Morgan, "Calibrating Cameras in Poor-conditioned Pitch-based Sports Games," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, Calgary, AB, 2018, pp. 1902-1906, DOI: 10.1109/ICASSP.2018.8461667.

[16] N. Homayounfar, S. Fidler and R. Urtasun, "Sports Field Localization via Deep Structured Models," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, Honolulu, HI, 2017, pp. 4012-4020, DOI: 10.1109/CVPR.2017.427.

[17] L. Citraro, P. Márquez-Neila, S. Savarè, V. Jayaram, C. Dubout, F. Renaut, A. Hasfura, H. B. Shitrit, P. Fua, "Real-time Camera Pose Estimation for Sports Fields," *Machine Vision and Applications*, vol. 31, no. 16, Mar. 2020, DOI: https://doi.org/10.1007/s00138-020-01064-7.

[18] X. Nie, S. Chen, and R. Hamid, "A Robust and Efficient Framework for Sports-Field Registration," in *Proc. IEEE/CVF Winter Conference on Applications of Computer Vision*, Virtual, 2021, pp. 1936-1944.
[19] K. He, G. Gkioxari, P. Dollar, and R. Girshick, "Mask R-CNN," in *Proc. IEEE International Conference on Computer Vision*, Venice, Italy, 2018, pp. 2961-2969.
[20] Y. Takezawa, M. Hasegawa, and S. Tabbone, "Robust Perspective Rectification of Camera-Captured Document Images," in *Proc*. IAPR International Conference on Document Analysis and Recognition, Kyoto, Japan, 2017, pp. 27-32, DOI: 10.1109/ICDAR.2017.345.

APPENDIX A

Vertex matching

Procedure 1: The detected vertices are clustered by their x and y coordinates, and the cluster numbers are assigned in ascending order of the coordinate values. Let c and r be the number of clusters for the x and y coordinates, respectively. The vertex labels are stored in the matrix $\mathbf{M}$ of size r × c according to each cluster number, where the cluster numbers for the x and y coordinates are the column and row numbers, respectively (e.g., see Fig. 7). The vertex labels of the court model are stored in an 8 × 6 or 6 × 8 matrix $\mathbf{M}_{\mathrm{m}}$: the size of $\mathbf{M}_{\mathrm{m}}$ is selected manually according to the court orientation. Zero is inserted into each empty element.

Procedure 2: The function *elasticMatching* is executed to obtain a matrix $\mathbf{M}_{\mathrm{Ex}}$ with the rows and columns stretched so that the non-zero elements of $\mathbf{M}$ match $\mathbf{M}_{\mathrm{m}}$. /* Because $\mathbf{M}$ is a matrix obtained by removing the rows and columns with 0 elements from $\mathbf{M}_{\mathrm{Ex}}$, $\mathbf{M}$ and $\mathbf{M}_{\mathrm{m}}$ correspond to each other. */

```
Function elasticMatching (M, M_m){
    matrix M_Ex
    for i = 1, 2 do{
        if M(0,0) ≠ 0 then M_Ex = ex00(M, M_m)
        else if M(0, c − 1) ≠ 0 then M_Ex = ex01(M, M_m)
        else if M(r − 1,0) ≠ 0 then M_Ex = ex10(M, M_m)
        else if M(r − 1, c − 1) ≠ 0 then M_Ex = ex11(M, M_m)
    }
    return M_Ex /* the matrix indicates matching result */
}

Function ex00(M, M_m){
    dynamic array v_0, v_1, v, w
    for y = 0, ⋯, M_m.rows − 1 do{
        for x = 0, ⋯, M_m.columns − 1 do{
            if M(0,0) = M_m(y, x) && y + r ≤ M_m.rows && x + c ≤
            M_m.columns then{
                createMatrix(M_Ex, (r + y)rows, (c + x)columns) /*
                initialized with zero */
                copyMatrix(M, M_Ex(range(y, y + r − 1), range(x, x + c −
                1)) /* copyMatrix(source, destination) */
                v_0.push(M_Ex)
                M_Ex.release
            }
    /* Column extension */
    for j = 0, ⋯, v_0.size − 1 do{
        v_1.push(v_0[j])
        for x = 1, ⋯, c − 1 do{
            for k = 0, ⋯, v_1.size − 1 do{
                if M(0, x) ≠ 0 then{
                    for all x_m such that: M(0, x) ≡ M_m(y_Ex, x_m(≥ x_Ex)),
                    where the (y_Ex, x_Ex) element of v_1[k] corresponds to the
                    (0, x) element of M.
                        copyMatrix(v_1[k], M_Ex)
                        /* insertColumns(destination, column, number of
                        columns) inserts the specified number of zero-padded
                        columns before the specified column. */
                        insertColumns(M_Ex, x_Ex, x_Ex − x_m)
                        w.push(M^Ex)
                        M^Ex.release
                }
                else{
                    for all x_m such that: M(y_nz, x) ≡ M_m(y_nz_Ex +
                    Δy(> 0), x_m(≥ x_Ex)), where the (y_nz, x) element of M
                    is the first nonzero element at the x column and the
                    (y_nz_Ex, x_Ex) element of v_1[k] corresponds to the
                    (y_nz, x) element of M.
                        copyMatrix(v_1[k], M_Ex)
                        insertColumns(M_Ex, x_Ex, x_Ex − x_m)
                        w.push(M_Ex)
                        M_Ex.release
                }
            }
            v_1.clear
            for k = 0, ⋯, w.size − 1 do{
                v_1.push(w[k])
            }
            w.clear
        }
        for k = 0, ⋯, v_1.size − 1 do{
            v.push(v_1[k])
        }
        v_1.clear
    }

    v_0.clear

    for j = 0, ⋯, v_2.size − 1 do{
        v_0.push(v[j])
    }
    v.clear

    /* Row extension */
    for j = 0, ⋯, v_0.size − 1 do{
        v_1.push(v_0[j])
        for y = 1, ⋯, r − 1 do{
            for k = 0, ⋯, v_1.size − 1 do{
                (omitted)
            }
        }
    }

    if v.size > 1 do
        filter(v) /* The image is homography transformed based on each
        correspondence that v[k] indicates, then the correspondence that
        maximizes the degree of coincidence between the color figures of
        the transformed image and the color figures of the court model is
        selected. Finally, copy it to v[0]. */

    return v[0]
}

Function ex01(M, M_m){
    (omitted)
    return v[0]
}

Function ex10(M, M_m){
    (omitted)
    return v[0]
}

Function ex11(M, M_m){
    (omitted)
    return v[0]
    }
```