Virtual advertising for volleyball videos

QiuRui Chen

ABSTRACT

This report proposes a framework of virtual advertisements insertion for volleyball matches. In this framework, connected component labeling, RANSAC, camera matrix, homography estimation and inverse warping are discussed.

KEYWORDS

Sports video advertising, Image advertising,Virtual advertising

1 INTRODUCTION

Nowadays, the sports video advertising draws a lot of attention due to its potential commercial benefits[1]. The advertisements can be divided into two categories: the physical advertisement and the virtual advertisement. The slogans appeared on the billboards or printed on the court in reality are physical advertisements. The virtual advertisements are inserted into sports videos by using computer aided blending techniques[2]. Usually virtual advertisements requires professional editors, but it is very labor-intensive and inefficient.

This report propose a framework of automatic virtual advertisements insertion in the volleyball videos. First we extract group feature of the court from the image and find the cross sections for the lines. Then camera matrix is calculated to find the connection between 3D world coordination to 2D image coordination. After that, homography is calculated to find the corresponding points from advertisement images to copy to court. Finally, inverse warping is performed to achieve the projection of advertisement onto the court.

2 METHODS AND MATERIALS

Figure 1 shows the work flow of the proposed framework for virtual ads insertion in the volleyball game scenes.

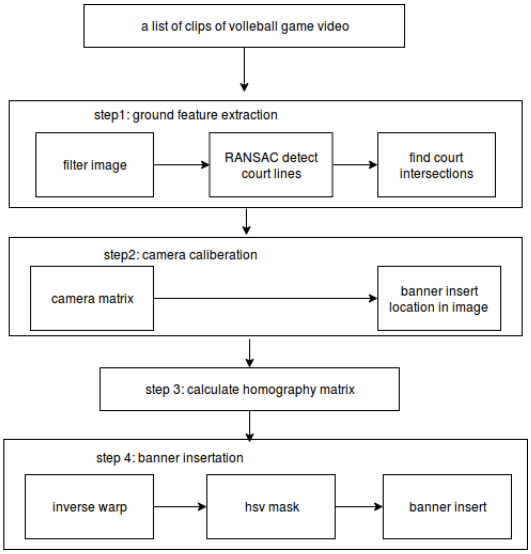

Figure 1: Work flow of the general process

2.1 Connected component labeling

Connect components labeling scans an image and groups its pixels into components based on pixel connectivity,i.e. all pixels in a connected component share similar pixel intensity values and are in some way connected with each other[3]. Extracting and labellings of various disjoint and connected components in an image is central to many automated image analysis applications[3]. Connected component labeling is based on pixel connectivity, which describe a relation between two or more pixels. For two pixels to be connected they have to fulfill certain conditions on the pixel brightness and spatial adjacency[3]. To formulate the adjacency criterion for connectivity, the notion of neighborhood is introduced. For a pixel p with the coordinates (x,y) the set of pixels given by:

$$N_4(p) = (x+1, y), (x-1, y), (x, y+1), (x, y-1) \quad (1)$$

is called its 4-neighbors. Its 8-neighbors are defined as:

$$N_8(p) = N_4 \cup (x+1, y+1), (x+1, y-1), (x-1, y+1), (x-1, y-1) \quad (2)$$

From above equations we can infer the definition for 4- and 8-connectivity: Two pixels p and q, both having values from a set V are 4-connected if q is from the set $N_4(p)$ and 8-connected if q is from $N_8(p)$ [4].

Then connected components can be labeled. The connected components labeling operator scans the image by moving along a row until it comes to a point p(where p denotes the pixel to be labeled at any stage in the scanning process) for which V=1. When this is true, it examines the four neighbors of p which have already been encountered in the scan (i.e. the neighbors (i) to the left of p, (ii) above it, and (iii and iv) the two upper diagonal terms). After completing the scan, the equivalent label pairs are sorted into equivalence classes and a unique label is assigned to each class[4].

Connected components labeling is very useful for filtering out objects in image. This method is applied based on below steps to select court.

- change image into binary image
- apply threshold
- get connected components in binary image
- filter out components with area large than 200, to keep large components
- do closing twice with disk shape
- get the connected components again
- filter out components with largest area and keep component with perimeter larger than 200, to filter out the Olympic rings component
- skeletonize the image

With above steps, candidate points are acquired.

2.2 RANSAC

Random sample consensus (RANSAC) is an iterative method to estimate parameters of a mathematical model from a set of observed data that contains outliers, when outliers are to be accorded no influence on the value of the estimates[5].

The candidate points include points at the top of images, they are "noise" ("outliners" in RANSAC) since they do not fit the model. The Hough transform will take those nose points into account which will lead wrong line model. So RANSAC is perfect line detection method for this project since RANSAC is a re-sampling technique that generates candidate solutions by using the minimum number observations (data points) required to estimate the underlying model parameters[6].

The basic algorithm is summarized in Algorithm 1.

Algorithm 1 RANSAC algorithm

```
ITERUM ← the number of iteration
THDIST ← the inliner distance threshold
THINLRRTIO * the amount of candidate points ← the inliner number threshold
for ITERUM do ▷ iterate ITERUM times
    sample ← randomly chose 2 points from PTS
    direction_vector ← secondpoint − firstpoint
    direction_vector ← a mod the Euclidean norm of the direction_vector
    unit_vector ← unit normal direction vector
    dist ← unit_vector * (all candidate points - first sample point)
    inliner_numbers ← count(dist < THDIST)
    if inliner_numbers < THINLRRTIO * the amount of candidate points then continue
    end if
    pca ← find principle components for all inliners
    theta ← use pca to find theta
    rho ← use pca to find rho
    for each saved theta and rho do get theta and rho with the most inlier
    end for
end for
```

Since there are 7 court lines, the process can be regarded as:

Iterate 7 times:

use RANSAC algorithm find one line;

delete in liners from candidate points;

After find seven lines, cross product is applied to find ten intersections of these court lines.

2.3 Camera matrix

Because court line and banner size(length: 20000mm, width: 50000mm) are known, camera matrix need to be calculated to find the connection between 3D world coordination to 2D image coordinate.

Here, twenty image frames are used to find camera matrix. For each image, ten image points and its corresponding points on world coordination are acquired.

Camera calibration is an important step toward getting a highly accurate representation of the real world in the captured images.

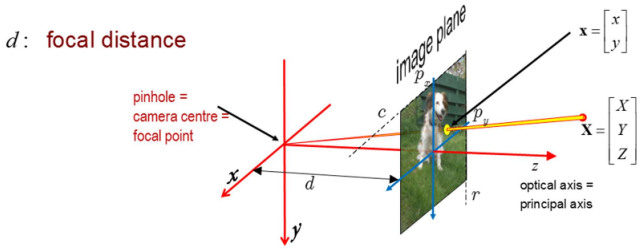

Figure 2: a pinhole camera model.

perspective projection:

$$c = \frac{Xd}{Z\Delta} + p_x r = \frac{Yd}{Z\Delta} + p_y \tag{3}$$

with

(r,c): Pixel coordinates or image coordinates

p_x, p_y: Principia point = image center

Δ : Pitch (distance between pixels)

often, d is expressed in Δ ,so that $\Delta = 1$

The figure 2 depicts a pinhole camera model. Both image points and world points need to translate into homogeneous form, that is add more dimension, get (u,v,1) and (x,y,z,1). Here, z = 0 is applied. For each point pairs, formula below is applied.

$$\alpha_n \begin{bmatrix} c_n \\ r_n \\ 1 \end{bmatrix} = \begin{bmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{21} & m_{22} & m_{23} & m_{24} \\ m_{31} & m_{32} & m_{33} & m_{34} \end{bmatrix} \begin{bmatrix} X_n \\ Y_n \\ X_n \\ 1 \end{bmatrix} \tag{4}$$

Above 3x4 M matrix equals intrinsic matrix multiply extrinsic matrix. The intrinsic matrix is below:

$$M_{in} = \begin{bmatrix} d_x & 0 & p_x \\ 0 & d_y & p_y \\ 0 & 0 & 1 \end{bmatrix} \tag{5}$$

and extrinsic matrix is below:

$$\begin{bmatrix} {}^c\mathrm{R}_w & {}^c\mathrm{t}_w \end{bmatrix} \tag{6}$$

where ${}^c\mathrm{R}_w$ is the rotation matrix that aligns the world coordinate system(WCS) to the camera coordinate system(CCS). And ${}^c\mathrm{t}_w$ is the origin of the WCS expressed in CCS.

From above equations and $N \geq 6$ point pairs, all parameters in camera matrix can be obtained.

2.4 Homography estimation

To project one image patch onto another, we need, for each point inside the inserting area in the video frame, to find the corresponding point from the logo image to copy over. In other words, we need to calculate the homography between the two image patches. This homography is a 3x3 matrix that satisfies the following:

$$X_{logo} \sim HX_{video} \tag{7}$$

or equivalently

$$\lambda X_{logo} = HX_{video} \tag{8}$$

where X_{logo} and X_{video} are homogeneous image coordinates from each patch and λ is some scaling constant.

To calculate the homography needed, for each image, the corners of the patches that need to warp between in the image are acquired.

A homography H maps a set of points $x = (x, y, 1)^T$ to another set of point $x' = (x', y', 1)^T$ up to a scalar:

$$x' \sim Hx \tag{9}$$

$$\lambda \begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{pmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \tag{10}$$

$$\lambda x' = h_{11}x + h_{12}y + h_{13} \tag{11}$$

$$\lambda y' = h_{21}x + h_{22}y + h_{23} \tag{12}$$

$$\lambda = h_{31}x + h_{32}y + h_{33} \tag{13}$$

To recover x' and y', divide equations 11 and 12 by 15:

$$x' = \frac{h_{11}x + h_{12}y + h_{13}}{h_{31}x + h_{32}y + h_{33}} \tag{14}$$

$$y' = \frac{h_{21}x + h_{22}y + h_{23}}{h_{31}x + h_{32}y + h_{33}} \tag{15}$$

Rearranging the terms above, a set of equations that is linear in the terms of H can be derived:

$$-h_{11}x - h_{12}y - h_{13} + h_{31}xx' + h_{32}yx' + h_{33}x' = 0$$
$$-h_{21}x - h_{22}y - h_{23} + h_{31}xy' + h_{32}yy' + h_{33}y' = 0$$

Finally, the above equations can rewrite as matrix equation:

$$\begin{pmatrix} a_x \\ a_y \end{pmatrix} h = 0 \tag{16}$$

With:

$$a_x = (-x, -y, -1, 0, 0, 0, xx', yx', x')$$
$$a_y = (0, 0, 0, -x, -y, -1, xy', yy', y')$$

$$h = (h_{11}, h_{12}, h_{13}, h_{21}, h_{22}, h_{23}, h_{31}, h_{32}, h_{33})^T \tag{17}$$

Matrix H has 8 degrees of freedom, and so, as each point gives 2 sets of equations,we will need 4 points to solve for h unique. So, given ten points (ten intersections point are founded by detected court-lines), we can generate vectors a_x and a_y for each and concatenate them together:

$$A = \begin{pmatrix} a_{x,1} \\ a_{y,1} \\ \vdots \\ a_{x,n} \\ a_{y,n} \end{pmatrix} \tag{18}$$

The problem is now:

$$Ah = 0 \tag{19}$$

However, due to noise in our measurements, there may not be an h such that Ah is exactly 0. instead, some small $\widetilde{\epsilon}$ could have:

$$Ah = \widetilde{\epsilon} \tag{20}$$

To solve this issue, the vector h can be founded to minimizes the norm of this $\widetilde{\epsilon}$. In order to achieve this goal, Singular Value Decomposition(SVD) is used, after get U,S,V vector, the last column of V will be the vector h. And then the 3x3 homography matrix can constructed by reshaping the 9x1 h vector.

2.5 Inverse warping

After finding the homography H, the next step is to wrap each image point using H to find its corresponding point in the logo, and then return set of corresponding points as a matrix.

This is done instead of the other way around is because if the inverse homography is calculated, and project all the logo points into the video frame, we will most likely have the case where multiple logo points project to one video frame pixel (due to rounding of the pixels), while other pixels may have no logo points at all. This would cause someplace in the video frame where no logo points are mapped. To avoid this, it requires to calculate the projection from video frame points to logo points to guarantee the every video frame gets a point from the logo.

Then each point in the video frame (x_{video}) is replaced with the corresponding point in the logo (x_{logo}) using the corresponding (x_{image},x_{logo}).

3 RESULTS

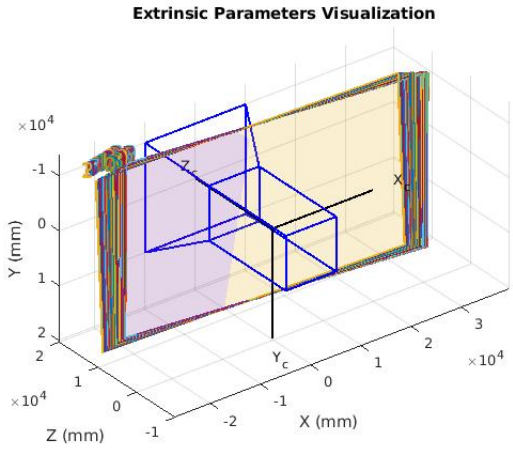

Figure 3: visualize multiple camera extrinsic parameters.

Figure3 shows the extrinsic parameters for samples clips which is used to calculate intrinsic matrix.

Figure 4 is one frame from video.

Figure 5 shows that change image into binary image and apply threshold to do further filter.

Figure 6 shows that after find 8-connected components in figure 5, remove relevant area according to their perimeter and area.

Figure 7 shows after filter, skeletonization is applied to get final points, which are candidate points for RANSAC method. Points at top of images are not belong to any court-lines, they are "outliers". Because they are quite far away from court-line dots, RANSAC would easy ignore them for each line detection.

Figure8 shows the detected court-lines after using RANSAC method.

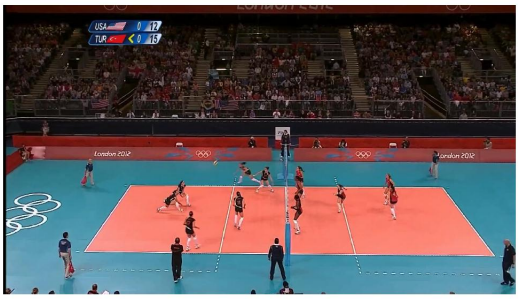

Figure 4: one frame clip of video.

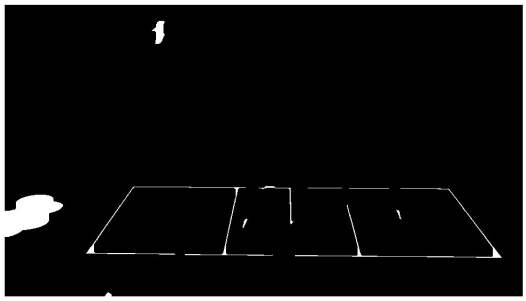

Figure 5: change into binary image

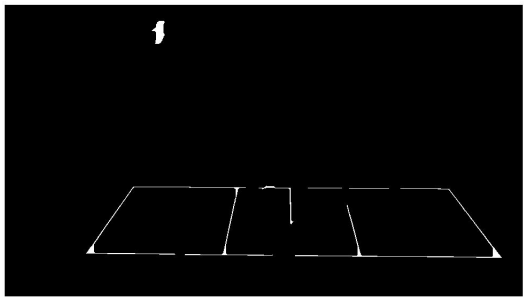

Figure 6: filter image

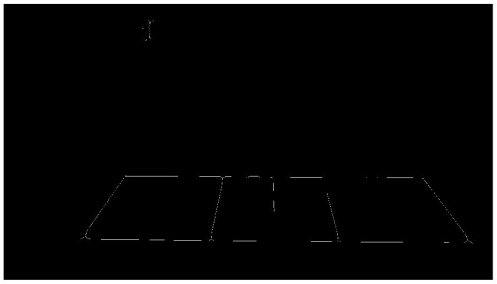

Figure 7: skeletonize

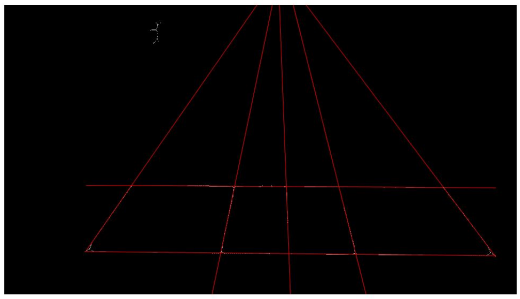

Figure 8: use RANSAC detect lines

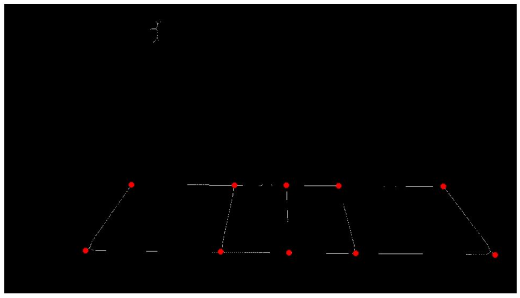

Figure 9: find intersection of court-lines

Figure 9 shows the corners found are intersection of court-lines. After finding intersections of 20 video frames and its corresponding points in WCS, intrinsic camera matrix can be acquired. Because video does not contain any zoom in or zoom out changes, intrinsic matrix retains all the same for all frames of this video. However, extrinsic matrix is quite different for every frames since every photo is taken from different angle, which means extrinsic matrix should be calculated for each frame.

Figure 10 is the court-line dimension this project makes use of. It also shows the banner location, that means WCS point for banner location is already known. With the camera matrix, this banner point in WCS can easily changed into 2D points on image plane.

Because the banner is only overlaid on the court, other things (running athletes or coaches) should be preserved. Here, HSV color representation is used, and some threshold is applied to create this mask.

Figure12 shows final image with banner inserted.

4 DISCUSSION

The research question is how to do placement and projection of a virtual advertisement.

The result shows that the virtual advertisement is basically done. But there are still many improvements can be taken. For key points

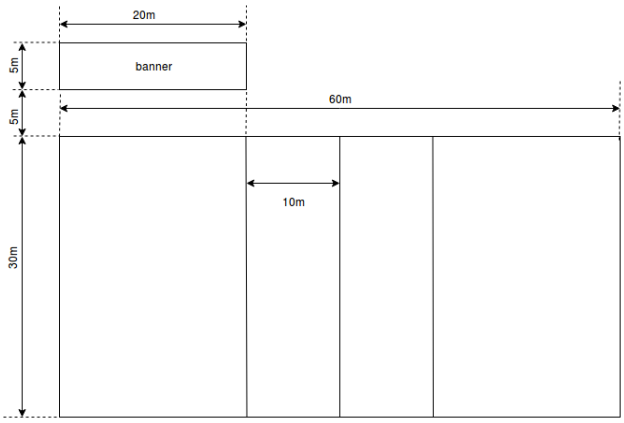

Figure 10: court dimensions and banner in actual world

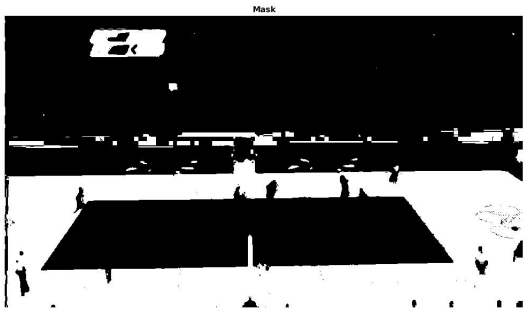

Figure 11: mask image based on color

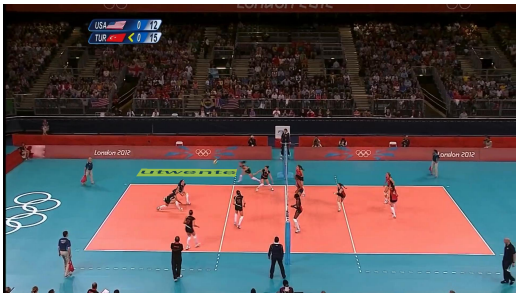

Figure 12: final projected image

detection, "noise"(points not belongs to court-lines) always exist in candidate points. It will influence RANSAC line detection if the amount of the candidate court-line points are small, which means "noise" points maybe will become "inliners".

For camera calibration, here, intrinsic matrix is fixed since the video is without the zoom lens. And in intrinsic matrix, shear constant is omitted. Also, nothing have done with lens distortion. Plus, fine tunning is also omitted after getting camera matrix. So, there are much more space we can do to improve camera matrix accuracy.

For final banner warp stage, mask created in order to protect moving athletes and coaches. Threshold parameter for HSV could be refined for more experiments.

And final 50% original image combined with 50% banner image are blended together to create transparent impact. There are better methods can be taken to let banner blend the background more smoothly.

There are some limitations for this project, say, image filter before applying RANSAC is constrained to this project. Also, before banner insertion, a mask is created to make sure banner only inserted on ground place. But the mask is based on blue ground color and this filter is not so precise.

Many literatures have studied the virtual advertisement. Color harmonization can be used to make banner blend with court environment better[2]. Some paper filter out ball-candidate with three properties of the fiidealfi ball image, which are size, ball-color range, and separation, this is used for intersection to create a mask to protect the ball[7]. One paper use vanishing point to define the homography transformation matrix[8]. It also do some line refinement(merging the segments lying on the same line and suppressing lines those are either close-by or too short) to find the best planar from building facade. Another paper tries to find the fitting planer by combination of RANSAC and Hough transform methods[9].

Regard to camera calibration, group-wise analysis is also used to get more stable camera matrix[7].

5 CONCLUSIONS

Virtual advertisement is popular and useful since it has huge commercial value. This report studies how to insert virtual advertisement on the volleyball game. With the help of camera calibration, the corresponding banner insert image points are founded by the reality banner inserting location. Banner can be successfully inserted into video with homography transformation.

A MATLAB RUN FILE SEQUENCE

To project the image, first run saveImage.m, then findCorners.m, finally run project_logo.m and video will be output in the folder.

B MATLAB SCRIPT FOR WARPING

```
function [ projected_img ] = inverse_warping( img_final, img_initial, pts_final, pts_initial )
% inverse_warping takes two images and a set of correspondences between
% them, and warps all the pts_initial in img_initial to the pts_final in
% img_final,
% image_final: video images; img_initial:logo_img
% pts_final:interior_pts;
% pts_initial:warped_logo_pts, video banner insertation corresponding points in banner
% round each element of X to the nearest integer greater than or equal to that element.

pts_final = ceil(pts_final);
pts_initial = ceil(pts_initial);

ind_final= sub2ind([size(img_final,1), size(img_final,2)],...
    pts_final(:,2),...
    pts_final(:,1));
ind_initial = sub2ind([size(img_initial,1) size(img_initial,2)],...
    pts_initial(:,2),...
    pts_initial(:,1));

% call createMask function to get the mask and the filtered image based on
% HSI color
    [BW,~] = createMask(img_final);

    [y,x] = find(~BW);
    mask =[x,y];

    mask_final= sub2ind([size(img_final,1), size(img_final,2)],...
        mask(:,2),...
        mask(:,1));
    [C,ia2,~] = intersect(ind_final,mask_final,'rows');
```

```
    ind_initial2 = ind_initial;
    ind_initial2 = ind_initial2(ia2,:);

    projected_img = img_final;
    for color = 1:3
        sub_img_final = img_final(:,:,color); % video image
        sub_img_initial = img_initial(:,:,color); % banner image
        sub_img_final(C) = sub_img_initial(ind_initial2)*0.5 + sub_img_final(C)*0.5;
        projected_img(:,:,color) = sub_img_final;
    end

end
```

REFERENCES

[1] N. Parameswaran J. Wang. Survey of sports video analysis: research issues and applications.

[2] Chia-Hu Chang, Kuei-Yi Hsieh, Ming-Che Chiang, and Ja-Ling Wu. Virtual spotlighted advertising for tennis videos. *Journal of Visual Communication and Image Representation*, 21(7):595 – 612, 2010.

[3] Ashley Walker Erik Wolfart Robert Fisher, Simon Perkins. Connected components labeling.

[4] Ashley Walker Erik Wolfart Robert Fisher, Simon Perkins. Pixel connectivity.

[5] Random sample consensus, 2017.

[6] Konstantinos G. Derpanis. Overview of the ransac algorithm.

[7] Xinguo Yu, Nianjuan Jiang, Loong-Fah Cheong, Hon Wai Leong, and Xin Yan. Automatic camera calibration of broadcast tennis video with applications to 3d virtual content insertion and ball detection and tracking. *Computer Vision and Image Understanding*, 113(5):643 – 652, 2009. Computer Vision Based Analysis in Sport Environments.

[8] Heather Yu Yu Huang, Qiang Hao. Virtual ads insertion in street building views for augmented reality. *International Conference on Image Processing*.

[9] Hitesh Shah and Subhasis Chaudhuri. Automated billboard insertion in video. *ACCV*.