Soccer Video Analytics

Abhinav Shukla, Shruti Gullapuram, Moneish Kumar

Motivation

The problem is an interesting application of a combination of image processing and semantic soccer logic in order to generate information about a broadcast soccer video.

When watching a game on TV, we can automatically perceive certain information based on what we see (goals, shots etc).

A state of the art analytics system should be able to process this information and also compute other information which is not very obvious (like possession, number of passes, shot to goal ratio etc)

Parts of the Project

1. Player/Ball Detection - *Frame processing, Background subtraction, player detection, ball detection, Mapping current view to actual pitch location*
2. Player/Ball Tracking - *Tracking player movement, Identifying a particular player, tracking ball movement.*
3. Event Detection - *Goals, Free kicks, penalties, corners, throws, Passes, Shots, Offsides.*

Statistic Generation - *Momentum, Possession, Player Specific*

Video used for the project

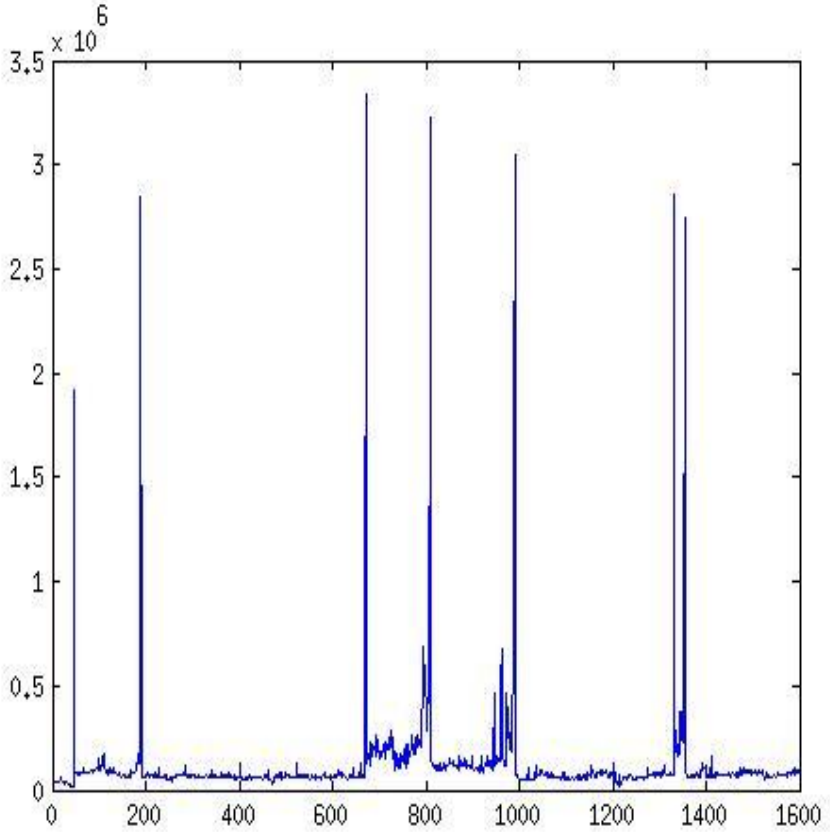
We used video from the 2014 FIFA World Cup semi final match between Brazil and Germany to conduct our experiments on.

We considered only the long view (from the central camera which is used most frequently during the normal course of the came).

Close-ups, camera changes, replay angles etc were not considered and the individual parts were done only in the long view which was determined using a histogram of differences.

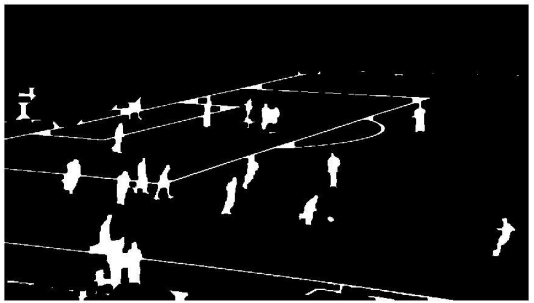
Frame Processing

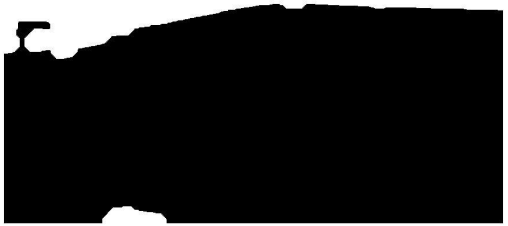
- Frames were processed to determine if we were in the long view or not (assuming we started out in the long view).
- Used a Histogram of Differences
- Sharp peaks in the histogram of differences indicates a hard cut (transition between long view and any other view)



Background Subtraction

- Done by first masking out the green colour of the field (using RGB) and then applying morphological operations (top-hat, transform, open, close) to further get the field, and players on the field.
- RGB based BG subtraction was used to completely separate the audience and the pitch.

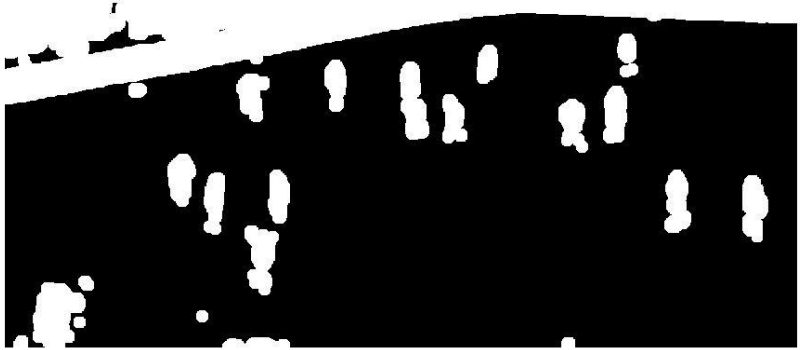




Background Subtraction

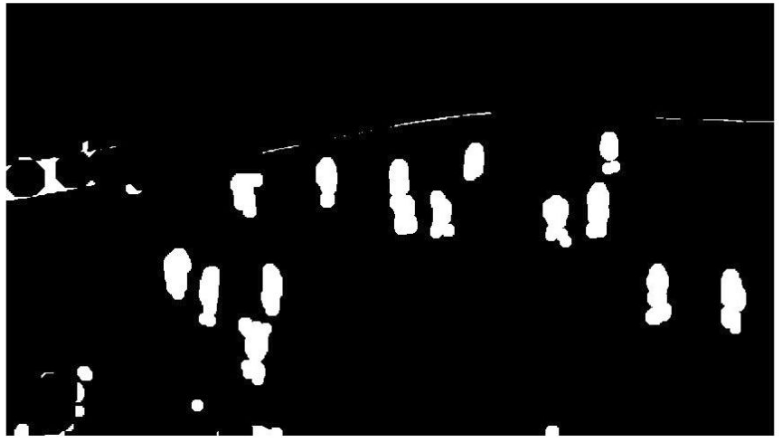
- Also used HSV to get a different kind of feature image.

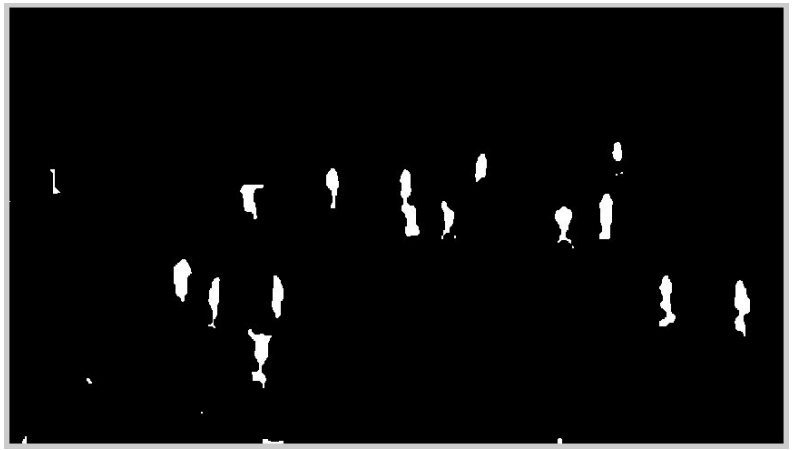
- HSV background subtraction was used to separate out the field lines and retain only the players to be used for the player detection step.



Player Detection

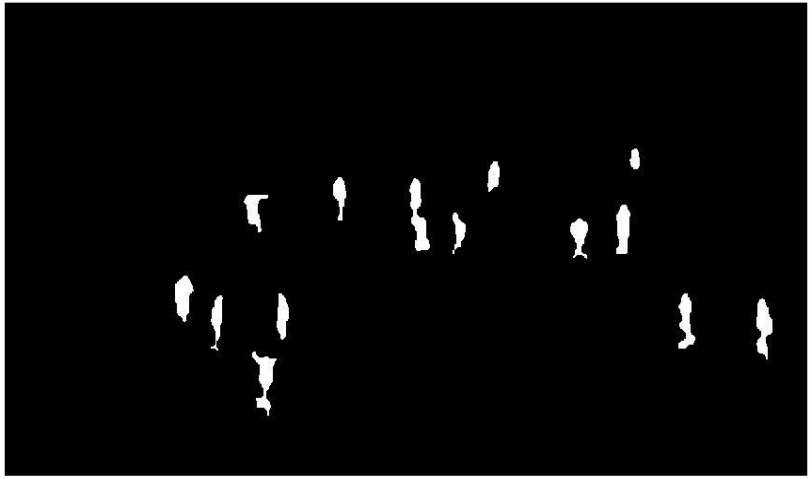
- Audience is subtracted from the HSV image using the audience mask
- The erosion morphological operation was applied to the result to remove any possible noise
- Further noise removed by finding connected components of area more than 200 as no player can be smaller than that





Player Detection

- In the final feature image, we find all such large connected components and draw a bounding box around each.
- The entity inside the bounding box is assumed to be the detected player.
- The detected players are then passed on to other parts of the pipeline for further processing.



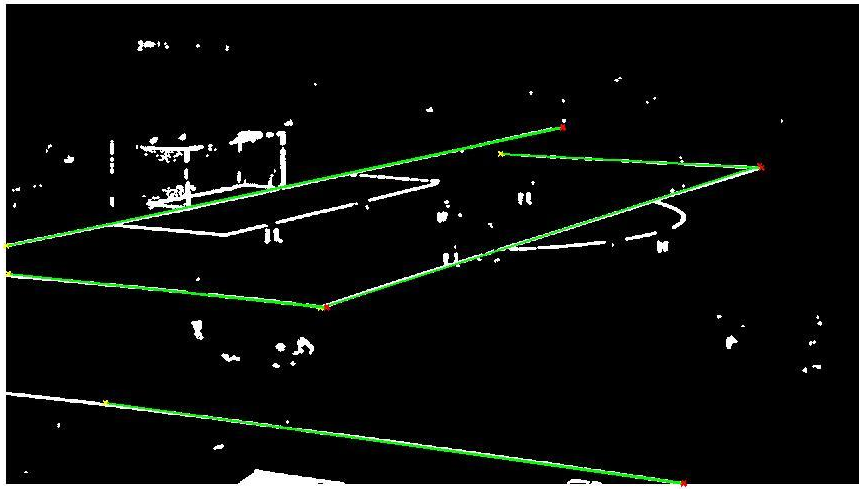
Mapping current view to actual pitch location

Mapping current view to actual pitch location

- Extracted the pitch. Using field lines to determine one of 13 key points.
- Using Hough transform on the thresholded feature image with high values of L in the LAB space used to find lines.

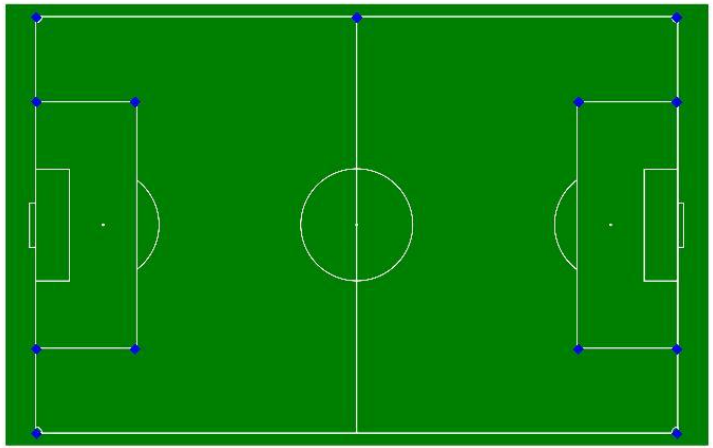
Mapping current view to pitch location

- Used morphological operations to get the required type of feature image on which Hough Transform was applied.
- Pitch location can be determined using the location of detected keypoints in the feature image.

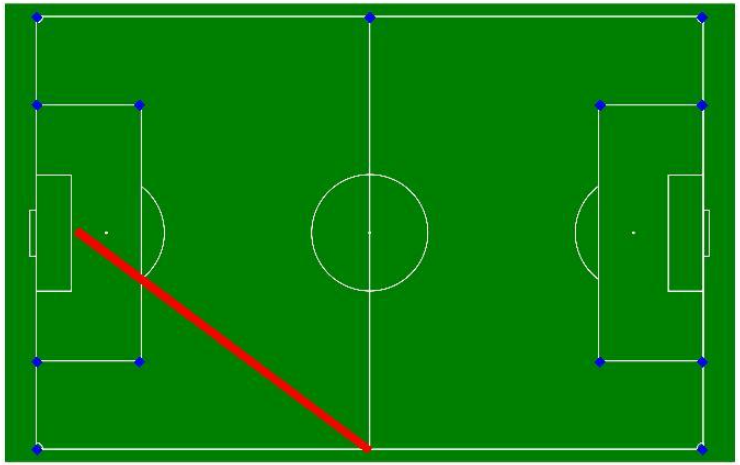


Mapping current view to pitch location

- These 13 key points were used. For images in which no points were detected, the change from the old angle was seen.
- If the camera was between the box and the centre (no points), depending on left or right, the angle was set to +22.5 or -22.5 which was a reasonably okay estimate.



Mapping current view to pitch location



Mapping current view to pitch location

- Tried stretching the contrast of the subtracted pitch in order to detect the angle of the mowing lines.
- Resultant feature images had too much noise.
- If this can be made to work, will accurately be able to map any view angle.

Tracking Player Movement

- For each connected component of the current frame, the distances between itself and the centroid of all the connected components of the previous frame are calculated. The component number that gives the minimum distance is stored in the connectivity map

val(:,:,1) =

Columns 1 through 9

122	74925	4901	28628	48698	102341	135481	188525	473705
72500	65	73989	42354	50580	64557	108193	237133	277285
2405	73666	100	15373	29353	73762	98564	140020	413482
23105	38898	12680	101	3917	25250	45568	102440	277874
42001	47060	25138	2221	37	11152	24634	75170	227588
92909	57762	67220	21125	9137	50	6676	62756	141674
126233	98162	92196	40113	21645	4410	100	36244	119866
172450	225817	126905	91892	68930	57881	31421	65	208133
450837	259874	394784	262769	215969	133130	117896	202816	82
421853	352468	352690	250297	200161	131368	88058	80002	66436
643345	541906	557000	425813	359453	262562	201344	178280	97682
886997	787194	782864	634169	553169	435650	352456	294416	201722
889300	922433	781453	675946	600100	510493	409625	270261	393237
1146865	1105256	1024642	878113	785725	660964	549842	424674	412704

Tracking player movement

- Each cell of a column maps the connected component of the current frame to the nearest component of the previous frame. Each column is a map between two consecutive frames.

	1	2	3	4	5	6	7	8	9	10
1	1	1	1	1	1	1	2	2	3	1
2	2	2	2	2	2	2	3	1	2	2
3	3	3	3	3	3	3	4	2	4	3
4	4	4	4	4	4	4	5	3	5	4
5	5	5	5	5	5	5	6	4	6	5
6	6	6	6	6	6	6	7	5	7	6
7	7	7	7	7	7	7	8	6	8	7
8	8	8	8	8	8	8	9	7	9	8
9	9	9	9	9	9	9	10	9	10	9
10	10	10	10	10	10	10	11	10	11	10
11	11	11	11	11	11	11	12	11	12	11
12	12	12	12	12	12	12	13	12	13	12
13	13	13	13	13	13	13	14	13	0	0
14	14	14	14	14	14	14	0	0	0	0

Tracking Player Movement

Identifying a particular player

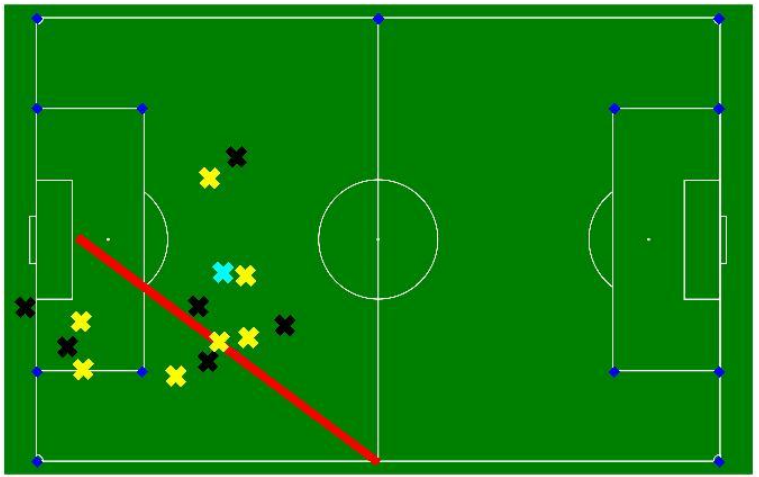
- From each of the detected subimages, the player was classified into one of three categories depending on the average color of the non-green pixels in the subimage.
- Euclidean distance from Brazil, German and Referee colors was used.

Identifying a particular player

- bra_color = [195 190 30];
- ger_color = [50 35 45];
- ref_color = [30 50 110];
- Trying to identify exactly which player proved to be difficult.
- Using weak cues like hair and boot color was problematic because they could not be detected accurately.

Identifying a particular player

- The location of each detected player was mapped to the pitch coordinates and plotted as shown.
- Rough estimate of player position over time to identify?

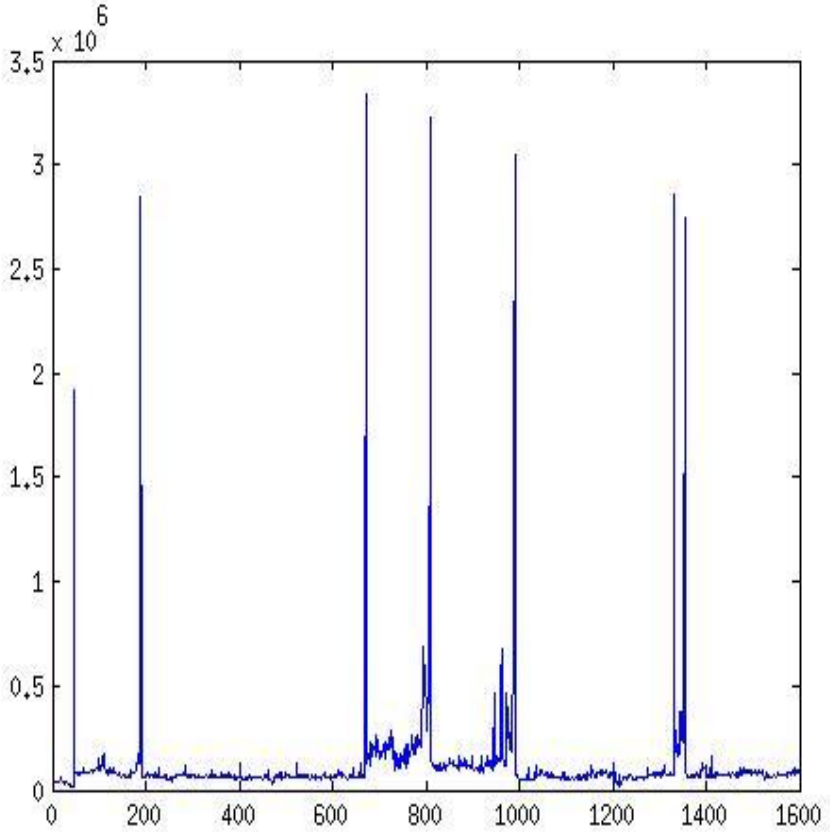


Tracking ball movement

- Done by first annotating a ball from any of the frame. Then using perform block matching (correlation) over the image which is spatially constrained by the location of the previous are where ball was detected.

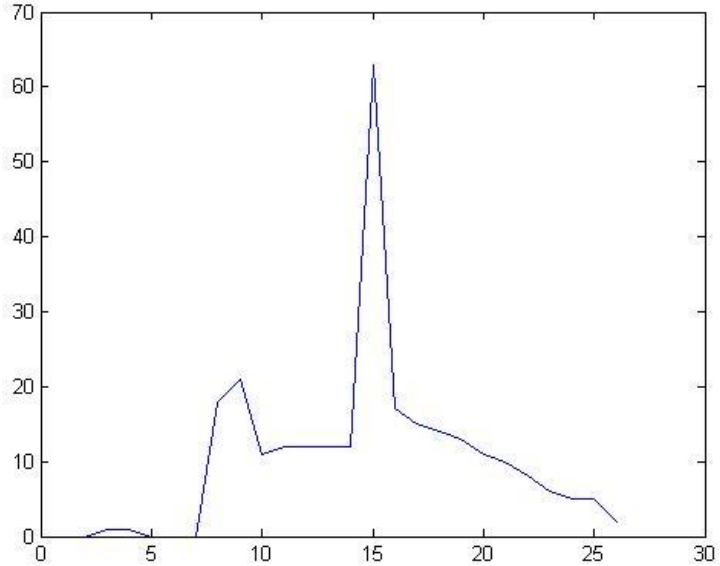
Goals

- Goals are immediately followed by replays which cause sharp changes in the histogram of differences, using which we can detect goals.
- Can differ between goals and simple view changes based on the length of time of the replay.



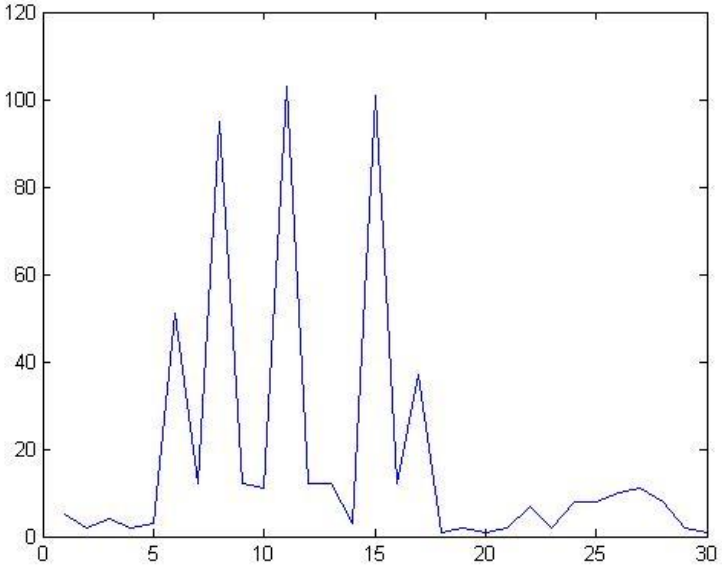
Passes

- A pass was detected as a sharp change in velocity of the ball.
- A peak in the plot of ball velocity vs frame number represents the instant that a pass was made.

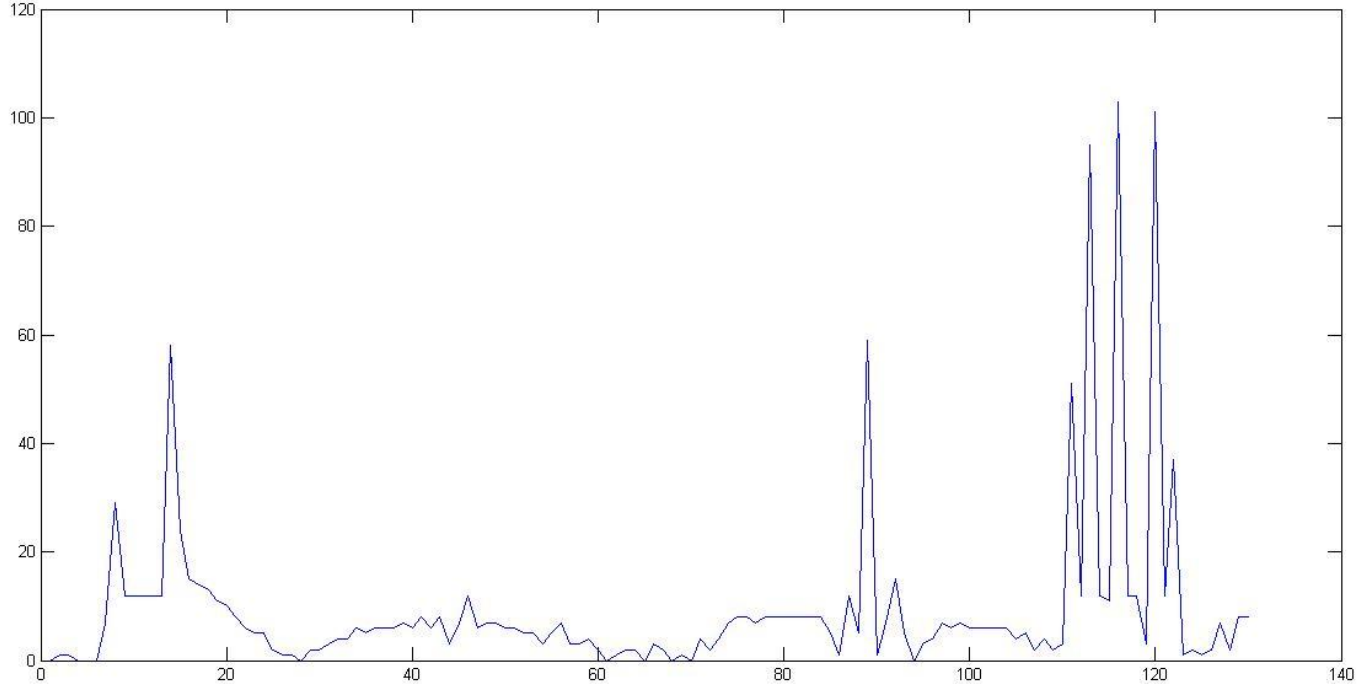


SHOTS

- A shot was detected as a sharp change in velocity of the ball.
- A shot is different from a pass in the fact that it has higher velocity and is near the goal.
- Multiple peaks due to uncompensated camera motion.

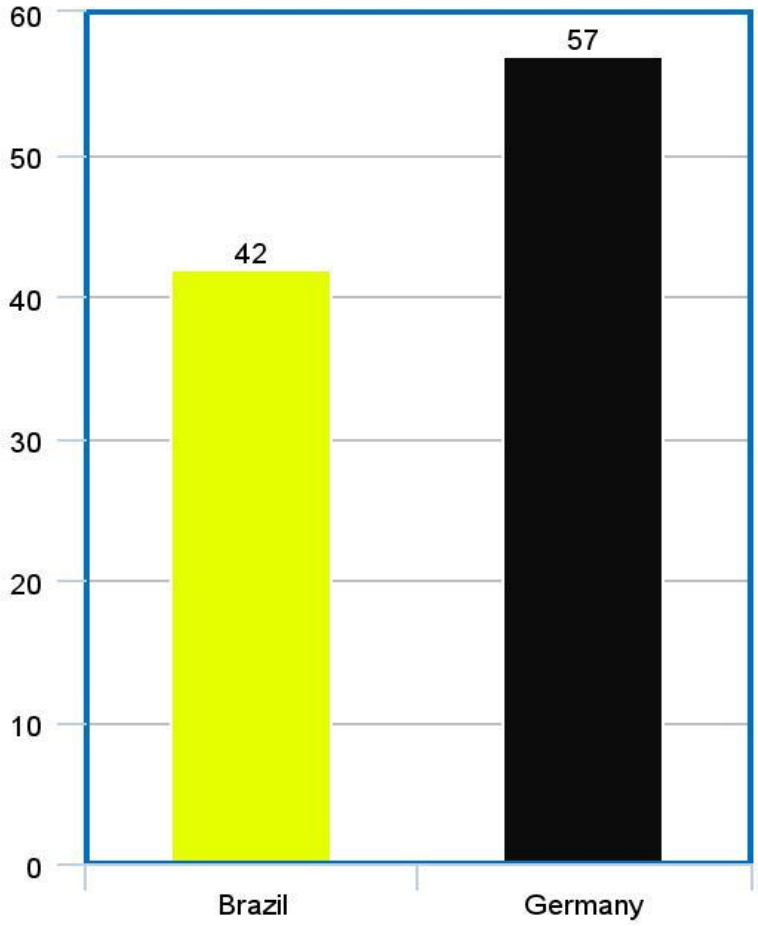


2 passes And a Shot Sequence



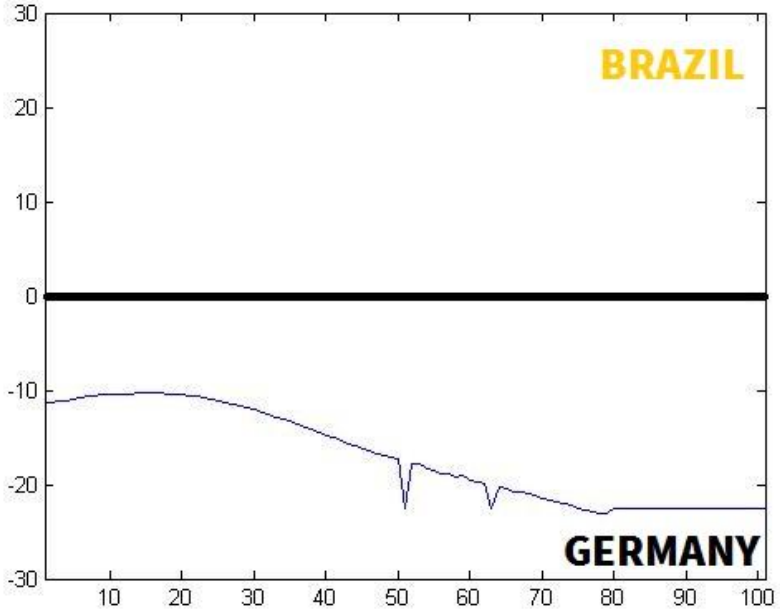
Ball possession

- We can successfully detect the location of the ball and of the players.
- Ball possession is decided based on the team of the closest player to the ball.
- Possession over time can be represented as a percentage as in conventional soccer games.



Game momentum

- A weak estimate of game momentum was done using the angle of view and which team had possession of the ball.
- e.g if Germany has ball possession and the camera is facing the Brazilian goal, the momentum is highly in favour of Germany.



A plot of current momentum (angle weighed by possession) gives the desired statistic.

Player Specific Statistics

- Player specific statistics could not be determined because an exact player could not be identified.

- If we were able to determine who an individual player was (would require training and classification), then we could associate relevant detected events with that particular player.

- Player specific heat-maps could not be generated either, due to the lack of identification.

Image Processing Techniques Used

- Color Histogram processing
- Extensive Morphological Operations
- Hough Transform
- Hopping in between different color spaces
- Thresholding at various places
- Template matching using NCC
- Histogram of Differences for View determination
- Geometry to convert between coordinate spaces
- Connectivity Map for Player Tracking

Possible Future Work

- Make the whole system more robust by intelligently detecting parameters from a random input video instead of having to specify individual parameters.

- Improve player classification from just splitting between two teams to actually being able to identify a particular player.

- Optimize the implementation which as of now is very slow.