HMM-based event detection using ball trajectories

Ibrahim Almajai

January 13, 2010

The problem

- Tennis event detection is important to interpret the game.

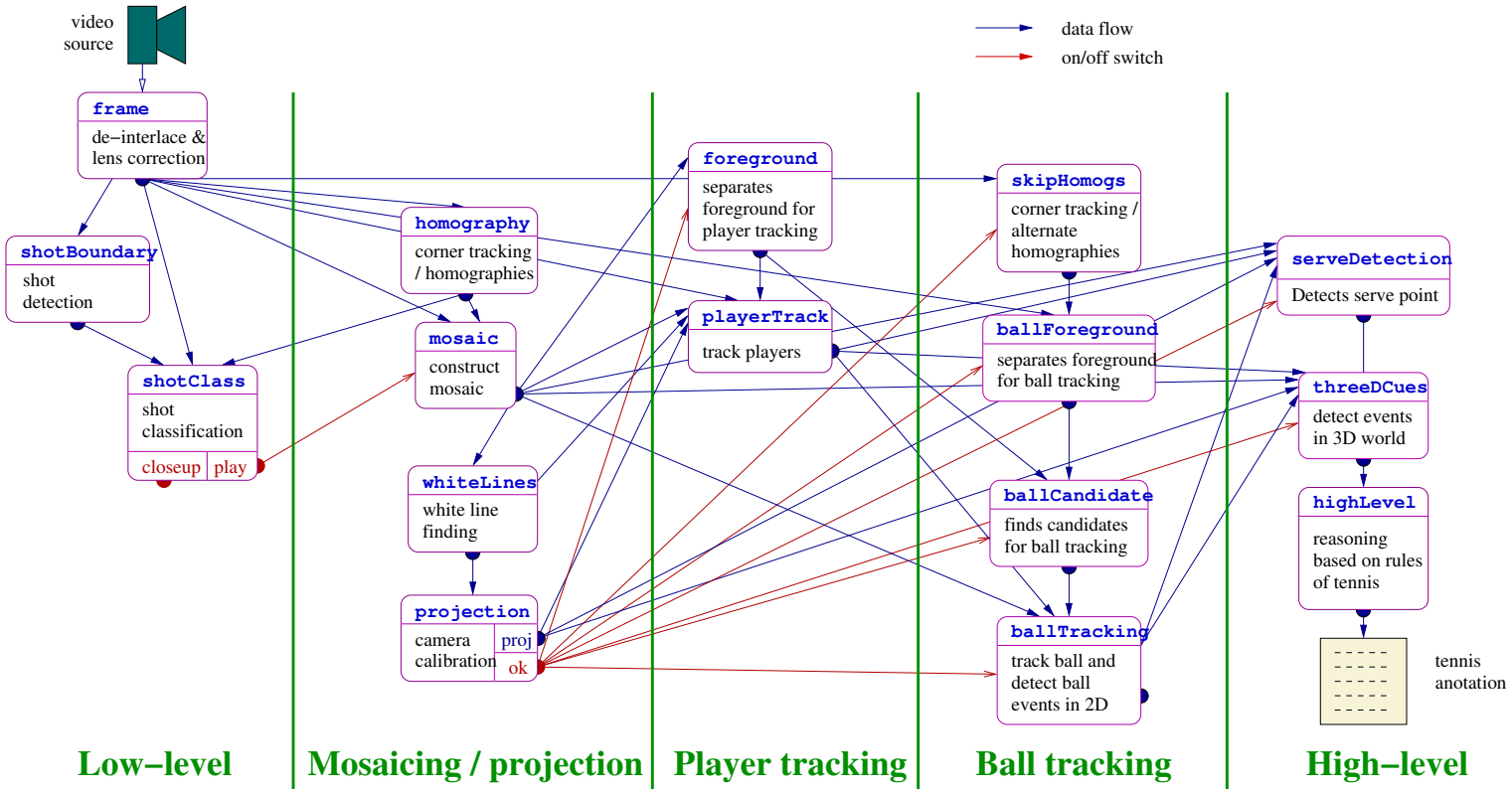
The problem

- Tennis event detection is important to interpret the game.
- It is prone to errors that can be more than the high level interpretation can tolerate.

The problem

- Tennis event detection is important to interpret the game.
- It is prone to errors that can be more than the high level interpretation can tolerate.
- Any improvement of the event detection accuracy is likely to decrease failures at the high level.

Tennis annotation system



Serve detection

A serve is detected when three conditions are met:

- A player is in the serve area.

Serve detection

A serve is detected when three conditions are met:

- A player is in the serve area.
- The player's body pose (contour) is associated with a serve.

Serve detection

A serve is detected when three conditions are met:

- A player is in the serve area.
- The player's body pose (contour) is associated with a serve.
- The ball is directly above the player.

Event detection

- The sudden changes in velocity of the ball trajectories are separately recorded as “ball events”.

Event detection

- The sudden changes in velocity of the ball trajectories are separately recorded as “ball events”.
- An algorithm of generalized edge-preserving signal smoothing is used to detect these keyevents

Event detection

- The sudden changes in velocity of the ball trajectories are separately recorded as "ball events".
- An algorithm of generalized edge-preserving signal smoothing is used to detect these keyevents
- Hit / bounce discrimination is achieved by considering how close is the player to the ball.

Event detection

- The sudden changes in velocity of the ball trajectories are separately recorded as “ball events”.
- An algorithm of generalized edge-preserving signal smoothing is used to detect these keyevents
- Hit / bounce discrimination is achieved by considering how close is the player to the ball.
- The court coordinates of the players and of ball events are projected onto the court surface model.

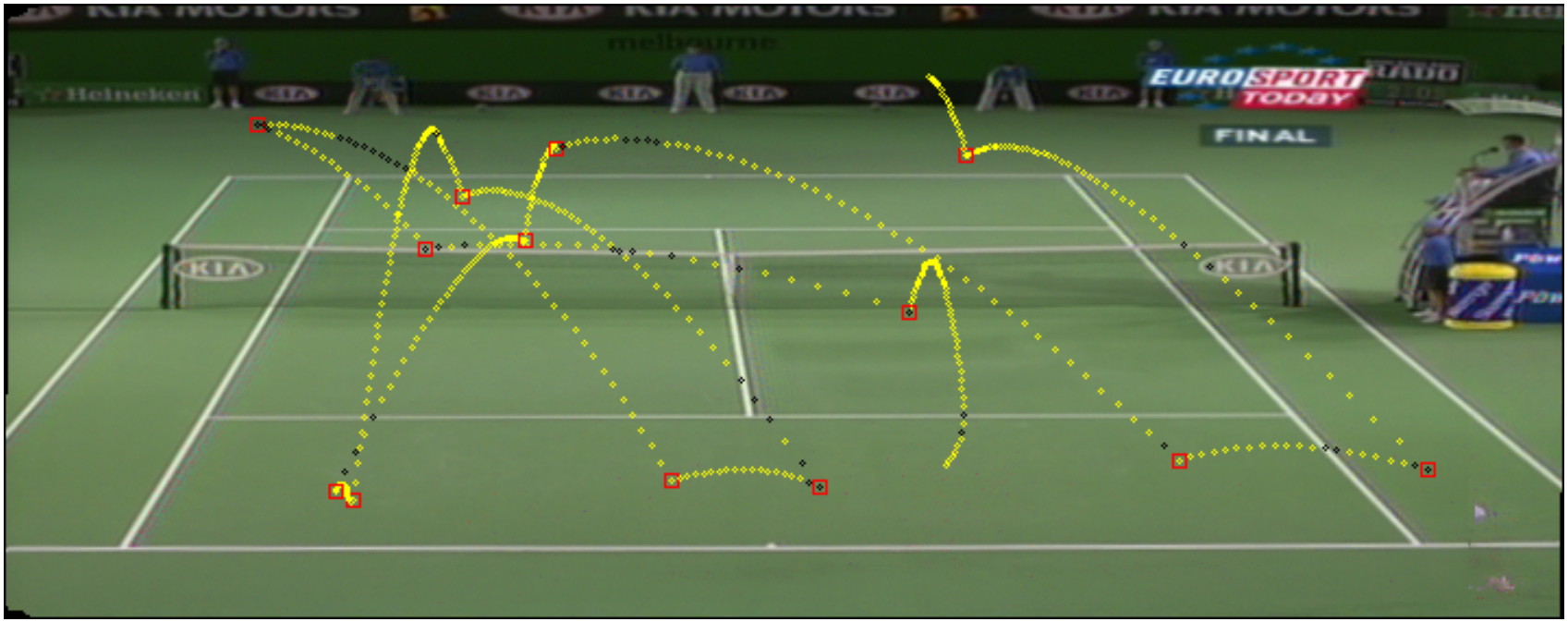
Event detection

- The sudden changes in velocity of the ball trajectories are separately recorded as “ball events”.
- An algorithm of generalized edge-preserving signal smoothing is used to detect these keyevents
- Hit / bounce discrimination is achieved by considering how close is the player to the ball.
- The court coordinates of the players and of ball events are projected onto the court surface model.
- The events are labelled according to the region of the court in which they occurred.

Event detection

- The sudden changes in velocity of the ball trajectories are separately recorded as "ball events".
- An algorithm of generalized edge-preserving signal smoothing is used to detect these keyevents
- Hit / bounce discrimination is achieved by considering how close is the player to the ball.
- The court coordinates of the players and of ball events are projected onto the court surface model.
- The events are labelled according to the region of the court in which they occurred.
- To avoid multiple instances of the same event in close succession, no event labels are permitted within 3 frames of each other.

Ball tracking result with event detection



Summary of tennis events used

Event	Description
SNR	Serve by Near player, Right Side
SNL	Serve by Near player, Left Side
SFR	Serve by Far player, Right Side
SFL	Serve by Far player, Left Side
BINSR	Bounce Inside Near player's Serve area on the Right
BINSL	Bounce Inside Near player's Serve area on the Left
BONS	Bounce Out of Near player's Serve area
BIFSR	Bounce Inside Far player's Serve area on the Right
BIFSL	Bounce Inside Far player's Serve area on the Left
BOFS	Bounce Out of Far player's Serve area
NET	Bounced on NET
HN	Hit by Near player
HF	Hit by Far player
BIN	Bounce Inside Near player's half court
BON	Bounce Outside Near player's half court
BIF	Bounce Inside Far player's half court
BOF	Bounce Outside Far player's half court

HMM for event detection

- HMMs provide an irresistible tool for modelling time-varying patterns such as the tracked ball trajectories.

HMM for event detection

- HMMs provide an irresistible tool for modelling time-varying patterns such as the tracked ball trajectories.
- It is proposed here to replace the current event detection mechanism.

HMM for event detection

- HMMs provide an irresistible tool for modelling time-varying patterns such as the tracked ball trajectories.
- It is proposed here to replace the current event detection mechanism.
- Models can be built for each type of event.

HMM for event detection

- HMMs provide an irresistible tool for modelling time-varying patterns such as the tracked ball trajectories.
- It is proposed here to replace the current event detection mechanism.
- Models can be built for each type of event.
- The ball position and its derivatives are the observations

$$\mathbf{p} = [p_x, p_y, p_x^{'}, p_y^{'}, p_x^{''}, p_y^{''}] \quad (1)$$

HMM for event detection

- HMMs provide an irresistible tool for modelling time-varying patterns such as the tracked ball trajectories.
- It is proposed here to replace the current event detection mechanism.
- Models can be built for each type of event.
- The ball position and its derivatives are the observations

$$\mathbf{p} = [p_x, p_y, p_x^{'}, p_y^{'}, p_x^{''}, p_y^{''}] \tag{1}$$

- The observation probability distribution at state j and time t, $b_j(o_t)$, is considered to be continuous and can be represented by M Gaussian mixtures:

$$b_j(o_t) = \sum_{m=1}^{M} c_{jm} N(o_t, \mu_{jm}, \Sigma_{jm}) \tag{2}$$

Experiments

- The dataset is composed of 65 play shots from a women's Australia Open tennis match.

Experiments

- The dataset is composed of 65 play shots from a women's Australia Open tennis match.
- A HMM of five emitting states of with different number of mixtures model per event (17 events) is trained using the HTK toolkit.

Experiments

- The dataset is composed of 65 play shots from a women's Australia Open tennis match.
- A HMM of five emitting states of with different number of mixtures model per event (17 events) is trained using the HTK toolkit.
- Training and test stages are done in a leave-one-out process because of the small size of the data.

Experiments

- The dataset is composed of 65 play shots from a women's Australia Open tennis match.
- A HMM of five emitting states of with different number of mixtures model per event (17 events) is trained using the HTK toolkit.
- Training and test stages are done in a leave-one-out process because of the small size of the data.
- First, the global mean and variance of all of the Gaussians in a given HMM are computed.

Experiments

- The dataset is composed of 65 play shots from a women's Australia Open tennis match.
- A HMM of five emitting states of with different number of mixtures model per event (17 events) is trained using the HTK toolkit.
- Training and test stages are done in a leave-one-out process because of the small size of the data.
- First, the global mean and variance of all of the Gaussians in a given HMM are computed.
- Following this flat start process (no given event boundaries), the HMMs are re-estimated using embedded Baum-Welch algorithm.

Experiments

- The dataset is composed of 65 play shots from a women's Australia Open tennis match.
- A HMM of five emitting states of with different number of mixtures model per event (17 events) is trained using the HTK toolkit.
- Training and test stages are done in a leave-one-out process because of the small size of the data.
- First, the global mean and variance of all of the Gaussians in a given HMM are computed.
- Following this flat start process (no given event boundaries), the HMMs are re-estimated using embedded Baum-Welch algorithm.
- Every time the number of Guassian mixtures is increased, one cycle of embedded re-estimation is applied.

HMM test

- Once models are trained, Viterbi decoding is used to detect events in an unseen shot with a grammar such as:
  ($E1 $E2 {$E3})
  where
  $E1 = SFR | SFL | SNR | SNL;
  $E2 = BIFSR | BIFSL | BINSR | BIFSL | NET | BONS | BOFS | <BIFSR HF> | <BIFSL HF> | <BINSR HN> | <BINSL HN> | <BOFS HF> | <BONS HN>
  $E3 = <HF BIN> | <HN BIF> | <HF BON> | <HN BOF> | HN | HF | BIF | BIN | BON | BOF;

HMM test

- Once models are trained, Viterbi decoding is used to detect events in an unseen shot with a grammar such as:
  ($E1 $E2 {$E3})
  where
  $E1 = SFR | SFL | SNR | SNL;
  $E2 = BIFSR | BIFSL | BINSR | BIFSL | NET | BONS | BOFS | <BIFSR HF> | <BIFSL HF> | <BINSR HN> | <BINSL HN> | <BOFS HF> | <BONS HN>
  $E3 = <HF BIN> | <HN BIF> | <HF BON> | <HN BOF> | HN | HF | BIF | BIN | BON | BOF;
- This basically forces the decoded sequence to start with a serve event followed by bounce in or out the serve area or net events. Then a variable number of rally hits and bounces expected before the tennis play shot is over.

Evaluation

- To analyse the accuracy of event detection by the the proposed and the previous method, their output is compared to the correct reference transcriptions.

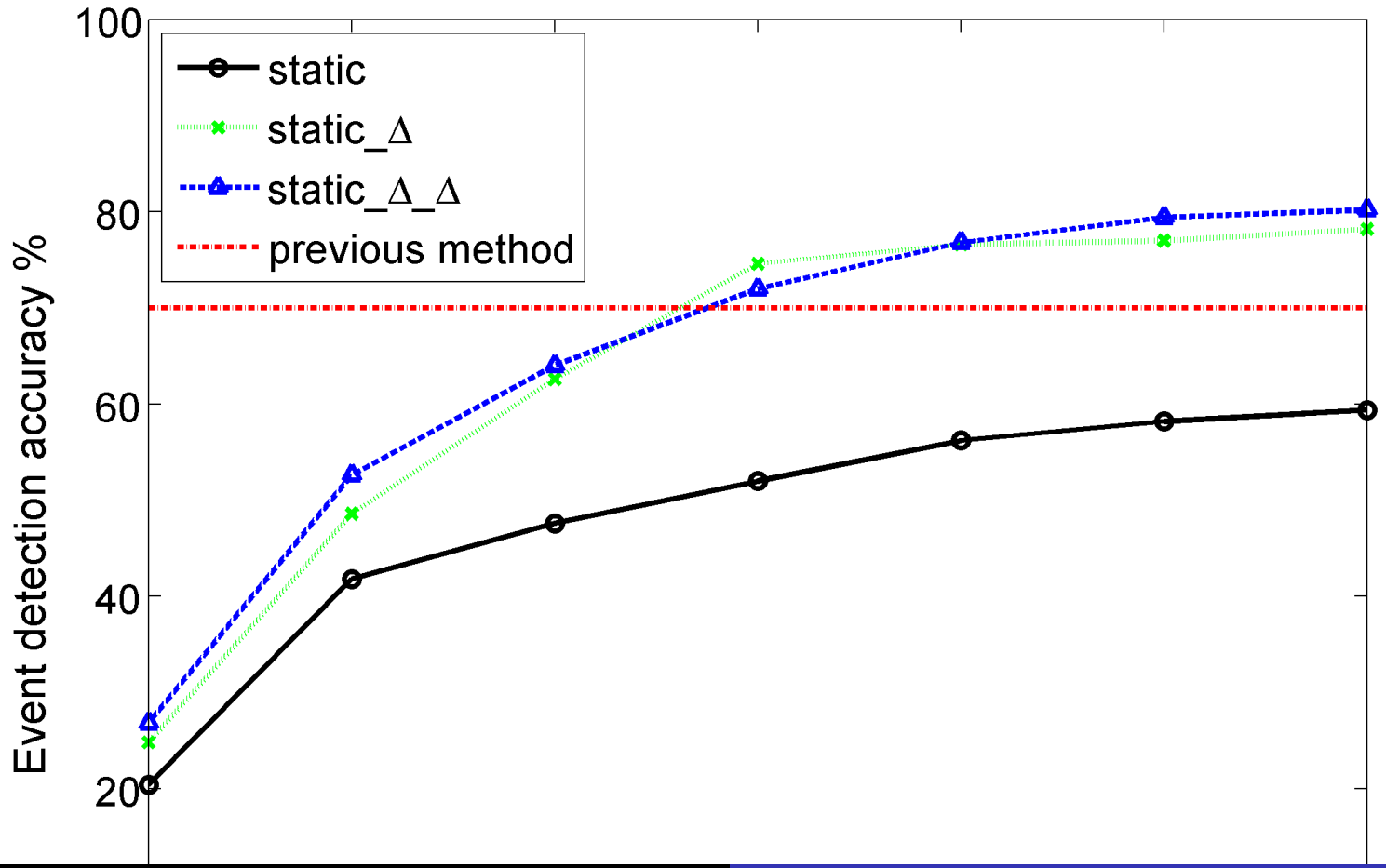
Evaluation

- To analyse the accuracy of event detection by the the proposed and the previous method, their output is compared to the correct reference transcriptions.
- This comparison is performed using dynamic programming to align the two transcriptions and then count substitution, deletion and insertion errors.

Evaluation

- To analyse the accuracy of event detection by the the proposed and the previous method, their output is compared to the correct reference transcriptions.
- This comparison is performed using dynamic programming to align the two transcriptions and then count substitution, deletion and insertion errors.
- An accuracy of 80.20% event detection was achieved by the Viterbi decoder compared to 70% of the previous method.

Event detection accuracy at different Guassian mixtures



Thank you