Object-Based Audio for Interactive Football Broadcast

Robert Oldfield

The University of Salford, Salford, UK

+44 161 2954001

r.g.oldfield@salford.ac.uk

Ben Shirley

The University of Salford, Salford, UK

Jens Spille

Technicolor, Hannover, Germany

Abstract An end-to-end AV broadcast system providing an immersive, interactive experience for live events is the development aim for the EU FP7 funded project, FascinatE. The project has developed real time audio object event detection and localisation, scene modelling and processing methods for multimedia data including 3D audio, which will allow users to navigate the event by creating their own unique user-defined scene. As part of the first implementation of the system a test shoot was carried out capturing a live Premier League football game and methods have been developed to detect, analyse, extract and localise salient audio events from a range of sensors and represent them within an audio scene in order to allow free navigation within the scene.

Keywords *Object-based audio; interactive broadcast; audio objects; audio feature extraction; 3D audio; football; salient audio events; spatial audio; audio extraction*

Introduction

Project background

The FascinatE project, funded under EU FP7, is developing an end-to-end AV broadcast system that will provide an immersive, interactive experience for live events. The system will allow users to create and navigate their own AV scene based on their preferences or by free pan, tilt and zoom control. As part of this project, techniques are being developed to provide new methods for capturing real-time audio events and embedding them within a modelled 3D scene to facilitate an interactive audio experience.

The FascinatE project is utilising Fraunhofer HHI's OmniCam (Schreer et al., 2013) to produce a high resolution 180° panorama that is stitched with the output of HD camera clusters, used to produce regions of interest (ROIs) that can be freely navigated or selected by the user. Methods have been developed for real-time live audio event detection, analysis, location and extraction in order to enable dynamic audio interaction for dynamically changing user-generated scenes. A modeled 3D audio scene has been defined allowing users to pan around and zoom

into the high resolution panoramic video with corresponding changes to the audio scene.

This user control involves a considerable paradigm shift for users whose role has then changed from that of *passive viewer* to *active participant* in creating the scene. For FascinatE audio this shift is analogous to moving away from traditional television broadcast and toward an experience more like a first person gaming environment where, for example, turning the character (visual scene rotation) leads to a corresponding rotation of the audio scene. This is unlike current practice of broadcasting sports events as described in section 1.2. Current practices utilise a static ambient surround sound regardless of camera movement together with other microphone feeds which are panned centrally and manually raised or lowered dependent on the location of the action on the pitch. The concepts of first person gaming have previously been linked with such an fotmat-agnostic/object-based approach to audio with scene representation formats such as VRML (Carey and Bell, 1997), however this paper presents a different approach to the representation of a format agnostic audio scene which will be described in section 2.5.

To facilitate this interactive audio experience a corresponding shift in production methods is required that involves developing event detection techniques and modeling of a 3D audio scene in such a way that new user requirements for free navigation can be satisfied. The potential for users to freely navigate a live event, and in particular, to be able to zoom *into* the video content introduces considerable challenges for the project; although the navigation and zooming is into a 2D video stream the impression for the user in zooming into a scene is that other events will move towards, around and even behind the viewing position. This implies that sound source locations must be clearly defined in 3D space relative to a 2D visual display.

In order to allow users to zoom into the scene and *beyond* sound source locations the project has had to develop methods to detect, analyse and auto-localise key audio events in addition to manipulating ambient 3D audio recordings. The audio event model developed for the project has been designed to function across a range of genres incorporating both sporting events and also other genres such as live music and theatre however its initial trial implementation documented here has been that of a live Premier League football game.

Television broadcast of football

Current methods for capturing sound at a live football match for 5.1 surround sound reproduction are based on associated production choices and values. Ambient sound such as crowd noise is generally recorded using a single SoundField® ambisonic microphone or stereo pair suspended high above the crowd. Sounds on the field of play are recorded using twelve highly directional shotgun microphones arranged around the pitch as shown in Fig. 1. The ambient crowd noise is reproduced as a static surround signal, unaffected by camera choice, pan and zoom and all pitch-side shotgun microphone signals are panned to the front centre of the mix. An engineer at the event is responsible for raising and lowering these microphone signals according to play so that only those

microphones picking up sound near to the action are active at a given point in time.

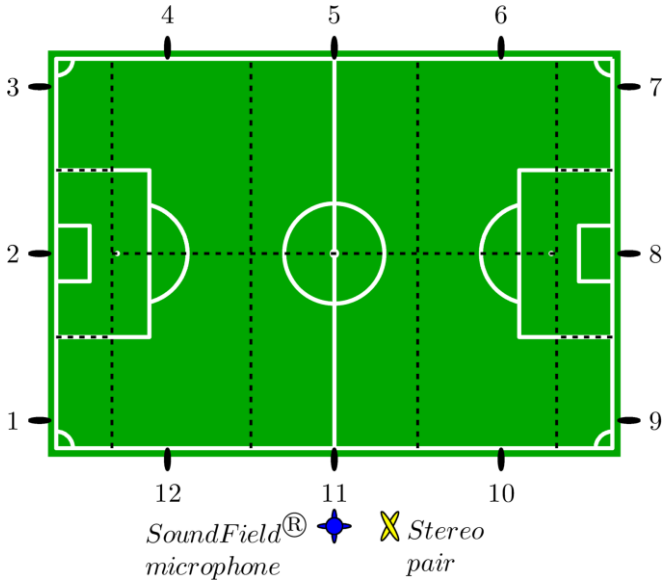

Fig. 1 Typical Microphone setup for and English Premier League match - the dotted lines mark the principle pick-up zones for each of the 12 shotgun microphones surrounding the pitch

Automatic mixing applications have been suggested in the literature (Cengarle et al., 2001), these techniques require that the engineer tracks the position of the on-pitch action using a remote device such as a tablet PC and the microphone signals are automatically mixed from this data. The object-based audio techniques presented in this paper could be used to realise an automatic mixing algorithm that would not require manual tracking. From the audio feeds alone, the algorithm detects when a significant on-pitch sound has occurred and adds the corresponding microphone feeds into the mix accordingly.

This paper is organised as follows. In section 2 an overview of object-based audio approaches is given including distinction between implicit and explicit audio objects, and definition of a new broadcast file/streaming format for the object-based audio scene. Section 3 describes the process of extracting the key on-pitch sounds of football as audio objects including object positioning. Section 4 describes the testing of the extraction algorithms; applications, improvements and conclusions are presented in sections 5, 6 and 7 respectively.

Object-based Audio

Background

Object-based audio operates on a different paradigm to channel-based systems. In channel-based audio, such as two channel stereo, 5.1, 7.1, 9.1, etc., the production (or the recording) is tailored for a specific loudspeaker configuration. In object-based systems each audio object, each sound, is defined not by its relationship to the loudspeakers and their placement but by a 3D coordinate location. In principal this means you can define your audio objects in production and render them across a wide variety of loudspeaker configurations, everything from conventional

5.1 and 7.1 to more immersive systems with a height component provided by loudspeakers above your head.

Object-based audio is considered an essential requirement of interactive systems in order to allow manipulation of the audio scene with reference to viewer/camera movements and to enable interaction with individual elements of an audiovisual scene. Bove Jr, (1995) and Watlington (Watlington and Bove Jr, 1997) suggest a computational framework for object-based encoding and suggest that in addition to being useful for compression, an object-based approach could open the door to interactive television content. These papers concentrate on video as being the biggest computation challenge owing to the high data rates involved, however an object-based approach to audio is also a necessary part of interactive and 3D media and poses many additional challenges.

One such challenge of implementing an object-based audio approach is that it puts new requirements on audio capture. Westner, (1998) suggests new methods for capturing the audio objects and identifies issues with separating acoustical sources in real-time. Kyriakakis, (1998) further emphasised the need for audio to play a part in the "suspension of disbelief" needed for 3D media systems and identifies both acoustical and technological factors that limit the capabilities and implementations of object-based 3D audio systems. Bove Jr, (1996) predicted that the benefits of such systems would include new production and post-production methods and include script based interaction with media objects. This paper also identified the difficulties inherent in analysis of audio scenes; correctly indicating that the rendering part of object-based audio is much more clearly understood and more easily carried out than the analysis. Various methods for separation of acoustic signals or blind source separation (BSS) have been proposed (Choi et al., 2005; Torkkola, 1999; Vincent et al., 2005) however the computing required for these higher order statistical analysis methods means that they are currently mainly unsuitable for real-time implementations such as that described in this paper.

Previous research has looked at the potential of using multimodal multimedia for high level event detection (such as a goal scored in a football game) (Chen et al., 2003; Wang et al., 2004) and also using audio only data (Kim et al., 2006). However this has been restricted to high level event detection such as detecting when goals are scored rather than low level events, e.g. when the ball has been kicked, and has been conducted in an offline situation for detection of football highlights.

Audio object description is also an open issue and several representations of audio objects have been proposed in the literature (Geier et al., 2010; Hoffmann et al., 2003; Peters, 2008; Pihkala and Lokki, 2003), one the most relevant here being the audio descriptors documented in the object-based multimedia standards MPEG-4 (MPEG., 1998). In this standard audio BIFS (BInary Format for Stream) are used to represent both natural and synthetic audio in a three dimensional *scene graph* (Scheirer et al., 1999) allowing implementation of 3D audio features into gaming, virtual environments and other interactive media applications (Lindsay and Herre, 2001). The MPEG-4 standard is extremely comprehensive and is perhaps hampered by its complexity which has not seen it widely adopted.

Audio Objects

Current broadcast standards do not use an objective audio approach but instead broadcast audio signals defined by the loudspeaker layout on which they are intended to be reproduced, so there are an equal number of audio channels as there are loudspeakers in the target reproduction system - two signals for two channel stereo, 6 channels for 5.1 surround etc. However the FascinatE project has at its root a *format agnostic* ethos; for audio this means that the audio must be captured and transmitted in such a way that it can be reproduced on any sound reproduction system, including large multi-channel systems such as ambisonics and wave field synthesis. This requires a shift in methodology from the current practice towards an object-based audio approach.

Broadcasting in this manner puts new requirements on the audio scene capture as most scenes contain many audio objects which may or may not be clearly defined. For this reason the complexity of the audio object capture depends on the type of audio source and also the capture system used. In this paper two types of audio objects are defined that need to be considered when capturing an audio scene.

Explicit

Explicit audio objects are objects that directly represent a sound source and have a clearly defined position within the coordinate system. This could include a sound source that is recorded at close proximity either by microphone or by a line audio signal and is placed, tracked or stationary with defined coordinates. Example: instruments close miked in a performance which are static and have little or no crosstalk from other sources, or the commentator at a football match.

Implicit

Implicit audio objects represent sound sources in a more indirect manner, these could include signals that are picked up by more distant microphone techniques or by microphone arrays where the source of sound may be derived from several recording devices. Example: in a football match the audio object describing the ball being kicked can be derived from one or more shotgun microphones around the pitch. In this instance the ball could not be tracked so both the content and position of the audio object need to be extracted from the available capture devices. This example is described in detail later in this paper.

Although both implicit and explicit objects need to be broadcast for most sports broadcasts, this paper focuses chiefly on the extraction of the position and content of implicit audio objects for sports broadcasting.

Broadcasting the Audio Objects

Once the content and position of the audio objects in the scene have been determined, they can be broadcast, together with the recorded ambient sound field, using a broadcast audio file format designed around object-based audio, to provide the user with a fully customisable audio scene

Requirements on an audio file/streaming format

The audio scene will generally consist of several audio tracks. These audio tracks contain single *audio objects* as well as *sound field descriptions*. The sound field

descriptions are represented in the well-known ambisonics format (Gerzon, 1973, 1985). Ambisonics is based on a spherical harmonic decomposition of a sound field which is then represented by a series of spherical harmonic coefficients, the ambisonic order defines the order of these coefficients and hence the spatial resolution of the recorded sound field. The number of tracks will vary over time as the content of the captured scene changes; new tracks can be 'born' (become active in the scene) and old tracks can 'die' (become inactive in the scene). A good example is a whistle-blow or ball-kick which will only be active in the scene at the extracted position for a short period of time. That track will 'live' for a few seconds so the broadcast reflects this by only broadcasting the necessary content for each period of time.

Over time the scene is split into audio frames, with a variable length as shown in Fig. 2.

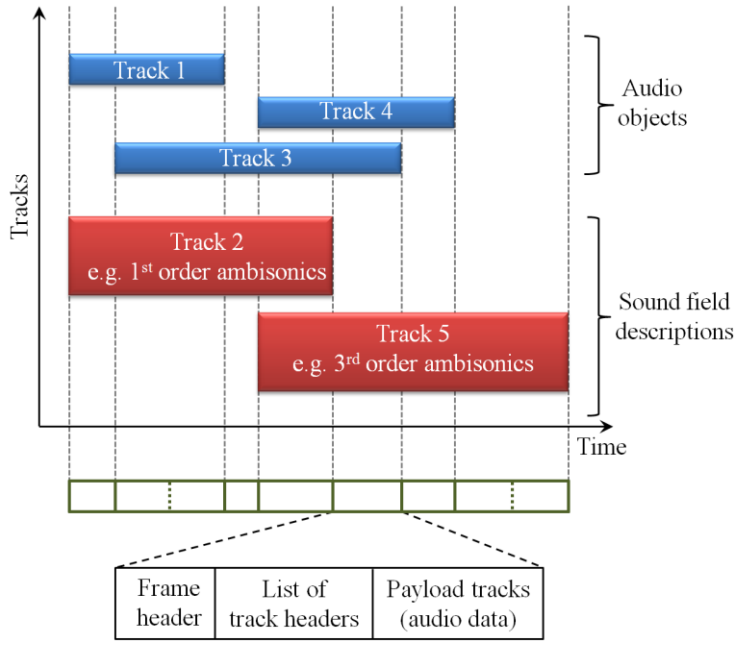

Fig. 2 Relation between audio tracks and frames

An audio frame contains information about the frame itself, information about the tracks and also of the payload (the audio data). A frame can be used as an access point to allow random access at some points in time. Instead of transmitting the whole frame length track by track, which would require a long buffer, the payload should also be split into smaller segments as depicted in Fig. 3.

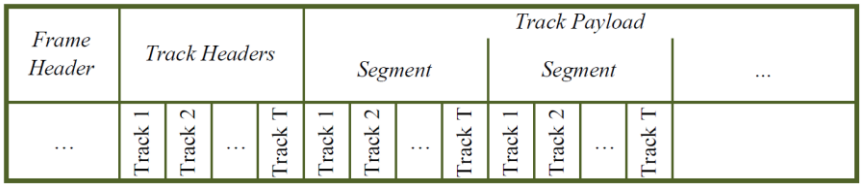

Fig. 3 Sequence of data packages per audio frame

The *Frame Header* has a unique frame number, a timestamp, the number of tracks, the sample rate, the number of samples per segment and potentially some room environment information could also be included. Each *Track Header*

requires a unique number, a timestamp offset, the track type (to distinguish between single Audio Objects and Sound Field tracks), the position and potentially a directional pattern using spherical harmonics coefficients or a VRML description (Carey and Bell, 1997). In case of a Sound Field track, parameters like 2D/3D, the Ambisonics order, the orientation of the Sound Field, the rotation, the bit depths, the coefficient order and the normalization method used (e.g. "Furse-Malham weights", "Schmidt Semi-Normalized" or 4π Normalized etc) should also be part of this header information.

Extracting Audio Objects

The algorithm presented here has been primarily developed for football applications but the concepts involved have application in other sports such as tennis, athletics, rugby etc. The algorithm ingests the audio feeds from the microphones surrounding the field of play in frames, analyses the content of each frame and assesses whether or not it contains a significant on-pitch audio event (OPAE). In the context of a football match, we define principally three categories of OPAE corresponding to ball-kicks, whistle-blows and players'/managers' communications, this paper concentrates on the detection and extraction of ball-kicks and whistle-blows. If the algorithm detects such an event in the audio frame, it determines that action is taking place in the vicinity of that microphone. For a standard broadcast scenario, this can be used to retrospectively add that microphone signal into the broadcast mix for the window of time in which the audio event occurred, this requires that the live be broadcast be delayed by approximately 1 second to allow for the processing time. For a broadcast scenario utilising an object-based audio approach, further processing can additionally be done to determine the location of the OPAE such that it can be extracted as an audio object with both content and location.

The methodology for extracting the audio events differs depending on the sport and type of OPAE that is to be detected however the principle of analysing the audio feed for salient features remains the same. An algorithmic approach has been applied here rather than using artificial intelligence techniques for the content extraction due to computation speed constraints on the system and the extremely large training data sets that would be needed especially for ball kicks where the sound changes significantly based on how hard the ball is kicked and which part of the boot/body the ball was hit with. Consequently the approach adopted here analyses the microphones for lower level audio characteristics for a very fast content extraction process.

Extracting ball-kicks

To successfully detect a ball-kick from the audio data, it is important to analyse the key characteristics of the audio generated by such a kick. Fig. 4b shows the spectrogram of a typical ball kick as recorded at a football match in the English Premier League.

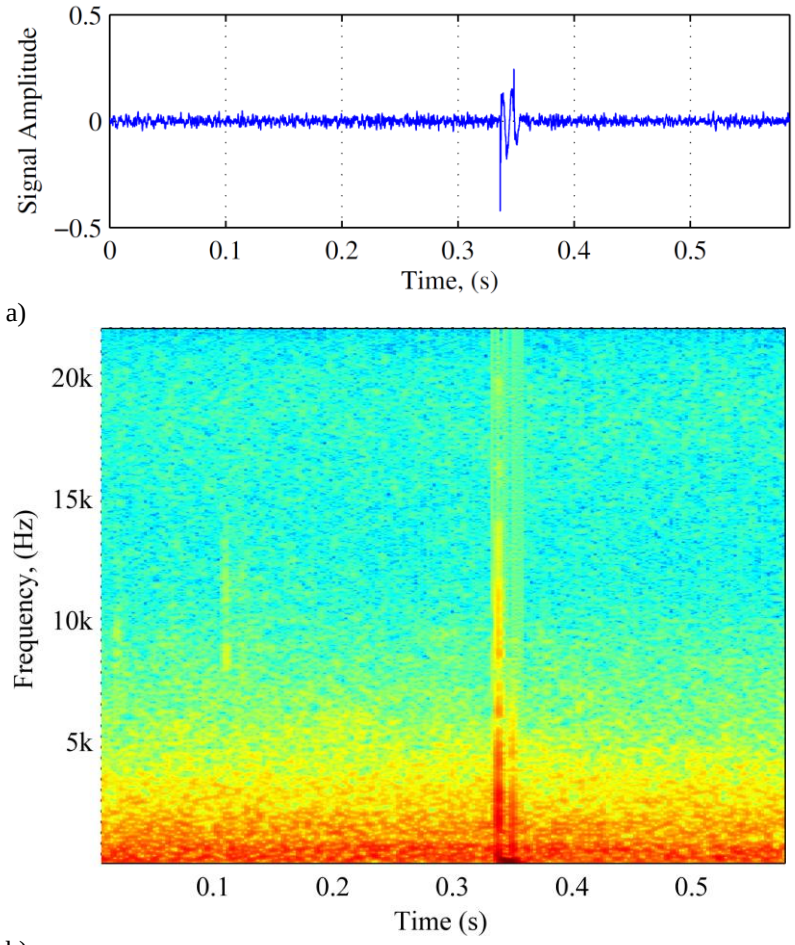

Fig. 4 a) Time history and b) spectrogram of a typical ball-kick from a live broadcast

As can be seen from Fig. 4b the kick of a football is characterised by broadband transient energy in the audio feed particularly at low frequencies. This is in contrast to the wash of crowd noise that contains few transients. This feature can be exploited to efficiently extract the audio from the microphone signal in real-time.

Using the recordings from the premier league match, 25 typical ball-kicks were analysed to look for the common signal characteristics that can be used to identify and extract the ball-kick from the broadcast microphone feed. These characteristics were used to fine tune the algorithm and identify signal thresholds for extraction. Fig. 4a shows the time history of a typical ball-kick which has a transient lasting ~25ms. In this case the ball-kick is significantly louder than the background noise, however there are many cases where this is not the case and visual inspection of the time history does not reveal the presence of a ball-kick, although the ball-kick is clearly audible. In this case, the sound of the kick can be extracted using other features of the audio signal.

The process for extracting the ball-kick can be described by the flow diagram in Fig. 5:

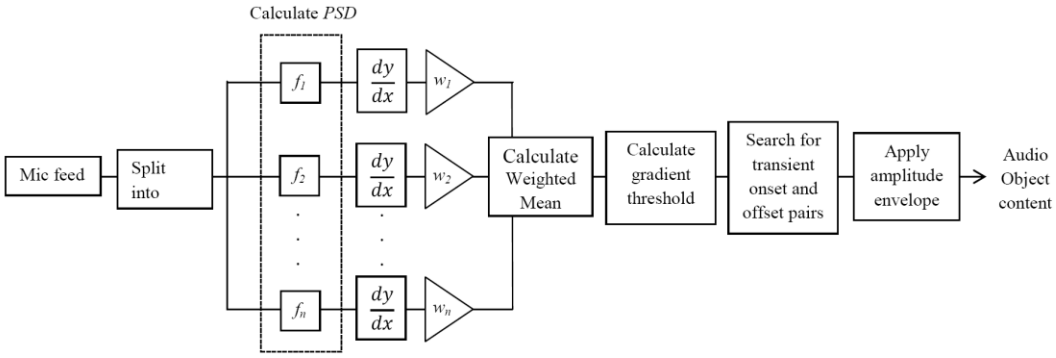

Fig. 5 Flow diagram of the ball-kick extraction process

Firstly the power spectral density of the input signal is computed at several analysis frequencies corresponding to the significant frequencies of an average ball-kick using the short-time Fourier transform (STFT) method. The frequencies chosen for the analysis here were 25, 30, 40, 50, 60, 80, 100, 125, 250 and 500Hz. The power spectral density is a measure of how the energy of a signal is distributed with frequency and is calculated from the square of the signal's Fourier transform as per equation 1:

$$PSD(\omega) = \left| \frac{1}{2\pi} \int_{-\infty}^{\infty} s(t) e^{-j\omega t} dt \right|^2, \qquad 1$$

This PSD function can be considered as the envelope of the signal in the given frequency band, thus the gradient of this function describes how quickly energy changes in that frequency band which illuminates points of the signal with transient energy. The gradient of this function is thus computed for all the analysis frequencies. A series of weights for each frequency are then applied to this gradient function giving greater or lesser preference to certain frequencies allowing fine-tuning of the search algorithm and for extracting transients with different characteristics for different scenarios such as the hit of a tennis ball where transients are still needed but the frequency content is very different. For the N weights, w_i a weighted mean is then taken to give a combined mean gradient, $\overline{G}$ as equation 2.

$$\overline{G} = \frac{\sum_{i=1}^{N} w_i G_i}{\sum_{i=1}^{N} w_i}, \qquad 2$$

The algorithm then searches through the samples in $\overline{G}$ to find both onset and offset transients. The average length of a ball-kick transient is 20ms although some have longer or shorter durations so the algorithm looks for onset and offset pairs that occur within 5-25ms of each other. This ensures that the algorithm doesn't detect global changes in sound pressure with a sharp transient onset or longer duration transients but only detects ball-kick transients. The gradient function is normalised to the median value of the signal in each frame and adaptive frame dependent threshold is derived to look for significant changes in transience. When this threshold is exceeded the algorithm has detected a ball-kick and the signal is multiplied by an amplitude envelope between the time interval of the detected ball-kick and the content of the audio object is determined. The onset time of the audio object must also be recorded and imbedded into the broadcast stream, as audio objects will become active and inactive at different times during the broadcast as described in section 2.5.

A further safeguard is put in place to prevent loud transient cheers from the crowd being erroneously extracted. This is done by looking at the sound power in the input channel. If the sound power exceeds a given value then the search algorithm is temporarily disabled. This is done for two reasons. Firstly, if the sound is very loud as is the case for a cheer when a goal is scored, or is about to be scored, the high level will cause a global increase in sound power which will highlight the sudden inclusion of the audio object into the mix (even after the attack and decay ramps have been applied). For standard audio object inclusion, this is not a problem as the audio from the ambient microphones picking up the crowd noise, mask the inclusion of the audio object (this also the case currently for a standard broadcast when the sound engineer adds the pitch microphones into the mix as described in section 1.2), but the masking of the audio object will be less effective if the audio object is very loud. The second difficulty is that if the extracted audio object contains a large amount of crowd noise and is mistakenly considered to be an OPAE, crowd noise will be positioned on the pitch, which when reproduced over a surround sound system will produce confusing auditory localisation cues and an incorrect audio scene. Future versions of the algorithm and new recording techniques however hope to alleviate these problems still further as described in section 6.1.

Enhancing ball-kicks

The sound of the ball-kicks can even be enhanced, bringing them out of the background noise for a better sound reproduction and listener experience, by multiplying the original signal by the absolute signal gradient over the analysis frequencies. This has the effect of increasing the transience of the signal which makes it perceptually more noticeable. Knowing the nature of the audio object, in this case a ball-kick, also means that a filter can be designed that further enhances this sound. The enhancing of on-pitch sounds is becoming more of a feature of modern broadcasts (Andrews, 2011) as consumers expect more immersion and a hyper real viewing experience more akin to a computer game.

Extracting whistle-blows

The technique for extracting the referee's whistle differs from that of extracting ball-kicks. The spectrogram of a typical blow from a referee's whistle as recorded during a broadcast of a live football match is shown in Fig. 6. For the duration of the whistle-blow (from $t \approx 0.7$s - 1.8s) it is characterised by its harmonic content, having a fundamental frequency of approximately 4 kHz. This feature can be exploited to detect and extract the whistle-blow by using the signal's cepstrum.

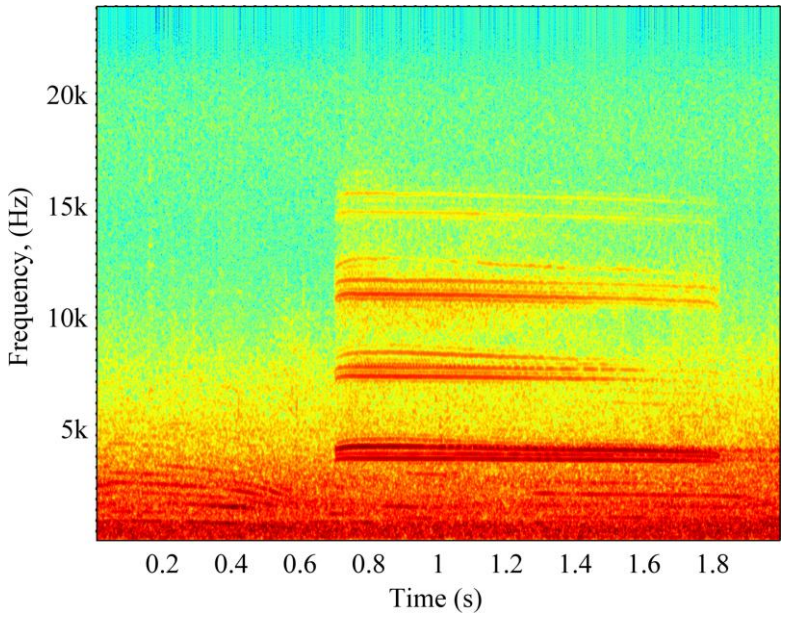

Fig. 6 A spectrogram of a typical referee's whistle-blow beginning at ~0.7s and finishing at ~1.8s

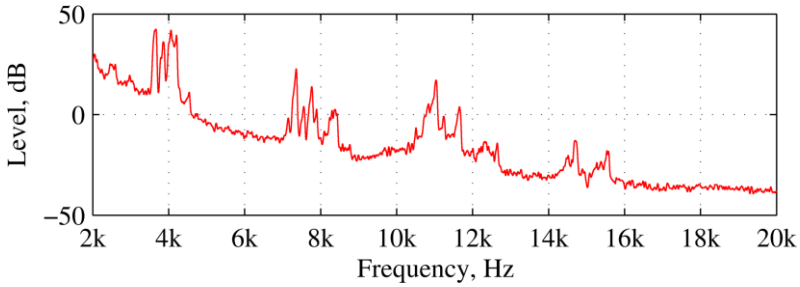

Fig. 7 Frequency spectrum of a typical referee's whistle blow - Note the harmonic content with ~4 kHz fundamental frequency

The cepstrum is often used in speech processing to determine the fundamental frequency of formants and for other scenarios where pitch detection is needed. The real cepstrum, c_x $c_x$is calculated by taking the inverse Fourier Transform of the magnitude of the natural logarithm of the source frequency response as per equation 3:

$$c_x = \frac{1}{2\pi}\int_{-\pi}^{\pi} \log|S(\omega)|e^{j\omega t}d\omega, \quad 3$$

Where $S(\omega)$ is the frequency spectrum of the source signal:

$$S(\omega) = \int_{-\infty}^{\infty} s(t)e^{-j\omega t}dt, \quad 4$$

As the cepstrum is calculated from the Fourier Transform of the frequency domain description of a signal, it highlights the periodic components in the frequency response, which hence equate to the signal's harmonic components. It is therefore ideally suited to detect a whistle-blow which, as seen from Fig. 6 and Fig. 7, is rich in harmonic content. If the input signal contains many harmonics, its spectrum will exhibit, peaks at the harmonic frequencies, whose spacing is

determined by the fundamental frequency of the signal. The more prominent these harmonics are in the signal, the greater the value of the cepstrum. The peak in the signals cepstrum will occur at a quefrency value, in temporal units, which is reciprocally related to the fundamental frequency. The cepstrum of a 0.2s section of a typical whistle-blow is shown in Fig. 8. The pink band in the plot corresponds to the analysis frequency range, i.e. the range of frequencies in which the fundamental frequency of the referee's whistle is likely to occur in (3 - 4.5 kHz). The extraction algorithm sums the cepstrum in this analysis band to determine when to extract a whistle-blow.

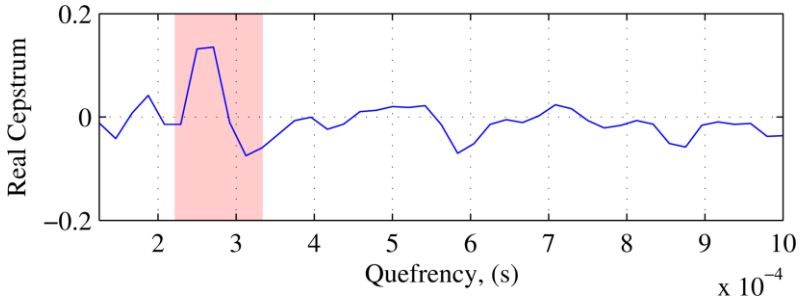

Fig. 8 Real cepstrum of a 0.25s chunk of a typical whistle-blow. The peak in the cepstrum at a quefrency of 2.7×10^{-4}s equates to a fundamental frequency of 3.7 kHz. The pink section shows the analysis frequency band for a typical whistle (3 - 4.5 kHz)

The method for whistle-blow extraction utilises a short-time cepstrum analysis with a signal frame length of 0.2s and with a frame overlap of 95%. A moving average of the summed value of the cepstrum in the analysis range is taken (as shown in Fig. 9) averaging the summed cepstrum value every 0.2s. The median value of the detected fundamental frequency for each 0.2s frame is also calculated. The algorithm looks for sections of the signal where the average summed cepstrum value is above the threshold value and the fundamental frequency falls within the frequency range of a typical whistle-blow (3 - 4.5 kHz).

If the above criteria are met, a whistle-blow has been detected and the microphone feed should be added into the mix for the corresponding time window and extracted as an audio object. In order for the onset and offset of the microphone signal to not be perceptually obvious an attack and decay ramp is applied with a 1.5s duration, the subsequent increase in level introduced with the addition of the microphone feed is not problematic as it is masked by the crowd noise from the ambient microphones as is the case for a standard broadcast when the engineer raises the level of shotgun microphones manually.

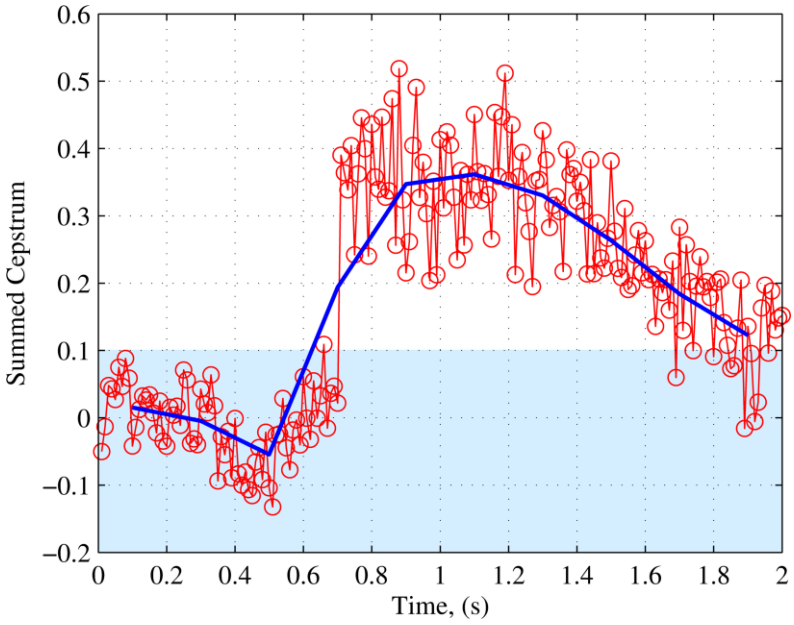

Fig. 9 Changes in the summed cepstrum with respect to time for the onset of a whistle-blow. The red line shows the summed cepstrum for each frame, the blue line shows the mean value for each 0.2s section of the signal. The light blue area shows the threshold value.

For the extraction of both ball-kicks and whistle-blows it would be possible to derive a receiver operating characteristic (ROC) curve (Metz, 1978) for the optimization of the level of the threshold, however this would only be of benefit to the specific recorded scenario as the required thresholds are liable to change based upon the given scene (i.e. it will change based on the stadium and the number of supporters in the stadium etc.). For a real implementation of the system the sound engineer would be given a sensitivity control which would enable a fine tuning of the extraction system, changing the threshold based on the specific scenario.

Extracting the object's location

Once the audio object content has been extracted using the techniques described above, the extracted content is compared with the signals from the remaining 11 microphone signals to determine if they contain the same signal. This is done by computing the cross correlation with the extracted source and the raw feeds from the other microphones in the time window of the audio object. If the cross correlation coefficient is high enough it is determined that the microphones are picking up the same signal and hence can be used to determine object's position. This step is necessary because the ball-kick or whistle-blow may only be a weak signal in the other microphones and consequently may not be detected by the extraction algorithm (especially if the sensitivity of the algorithm is reduced to avoid false detections), but if a significant amount of the audio source is in the microphone signal at even a very low level, it can still be used to position the source on the pitch.

The source positioning is based on a time delay estimation technique (Carter, 1993). The cross correlation between microphone pairs is computed and hence the delay between the received signals is calculated. The cross correlation can be performed using several techniques (Barsanti and Tummala, 2003; Cheng and Tjhung, 2003; Grennberg, Anders and Sandell, 1994; Jakobsson et al., 1998;

Knapp and Carter, 1976; Zongchuang et al., 2002); here we use generalised cross correlation coefficient (GCC) with the phase transform (PHAT) pre-filter (Knapp and Carter, 1976) as it is more robust to signals with a lower signal to noise ratio.

To find the time delay, D between two microphone signals $x_1(t)$ $x_1(t)$and $x_2(t)$ the GCC is computed:

$$R_{x_1x_2}(\tau) = \int_{-\infty}^{\infty} \psi_{phat}(f) G_{x_1x_2}(f) e^{j2\pi f\tau} df \,, \quad 5$$

where the phase transform pre-filter ψ_{phat} is given by

$$\psi_{phat}(f) = \frac{1}{\left|G_{x_1x_2}(f)\right|} \,, 6$$

and $G_{x_1x_2}(f)$is the cross power spectral density between the two signals. The time offset, τ at which this function $R_{x_1x_2}(\tau)$ is maximum gives the estimate of the relative delay between signals $x_1(t)$ $x_1(t)$and $x_2(t)$. This TDOA allows the calculation of the relative distances from the source to each microphones, ΔR as given by equation 7 where the geometry is given by Fig. 10.

$$R_1 = R_2 + \Delta R = R_2 + c\Delta t \,, \quad 7$$

Where Δt is the time-difference-of-arrival between the sound arriving at *Mic 1* and *Mic 2* and c is the speed of sound in the medium.

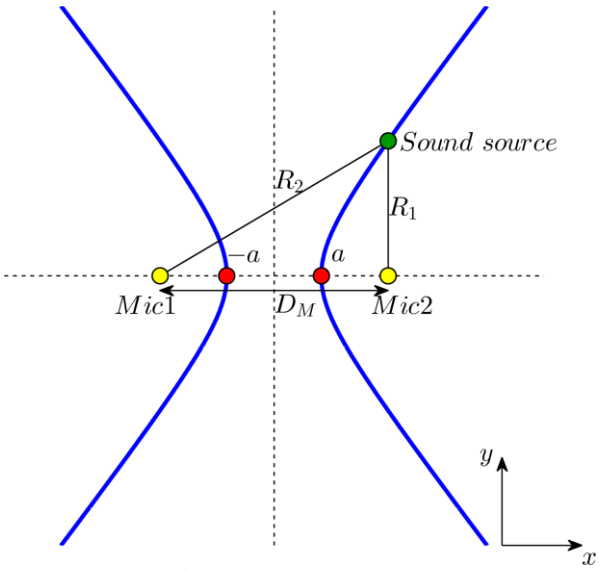

Fig. 10 Source geometry

There are many source positions that satisfy equation 7, plotting these possible positions reveals that they can be located along a hyperbola between the two microphones as a hyperbola is given by a set of all points in a plane such that the difference of the distances from two fixed points (foci) is constant. In this case the foci are the microphone positions and the source will be located on either the right hand or left hand hyperbolic path shown in Fig. 10 depending on which microphone received the sound first.

It can be shown that for a hyperbola the relative distances between R_1 and R_2 is given as

$$\Delta R = R_2 - R_1 = 2a\,, \quad 8$$

Where a is the absolute value along the x-axis from the origin of the vertices of the hyperbola. The equation for the hyperbola is then given as

$$\frac{x^2}{a^2} - \frac{y^2}{b^2} = 1, \quad 9$$

Where b is derived from the asymptotes of the hyperbola as:

$$b = \sqrt{\frac{D_M{}^2}{4} - a^2}\,, \qquad 10$$

Thus a hyperbola can be plotted, taking the positive solution of x along which the source will be situated. It should be noted that this derivation constitutes a specific geometry where the two microphones are located along the x-axis and equally spaced around the origin, thus for microphones not at these positions a coordinate transform is required to plot the correct hyperbola.

If there are two or more microphone pairs containing the same source signal, the source can be accurately positioned in 2D space by finding the point of intersection of the computed hyperbolae. If there are many hyperbolae, the points of intersection are averaged to give an accurate estimation of the source position. This can be seen from the measured data in Fig. 13. This approach has been chosen over more complicated optimization methods for source localisation (Benesty., 2000; Do et al., 2007; Silverman et al., 2005) due to the large search area that would be involved here; as such this constitutes a very computational efficient method which can be carried out in real-time.

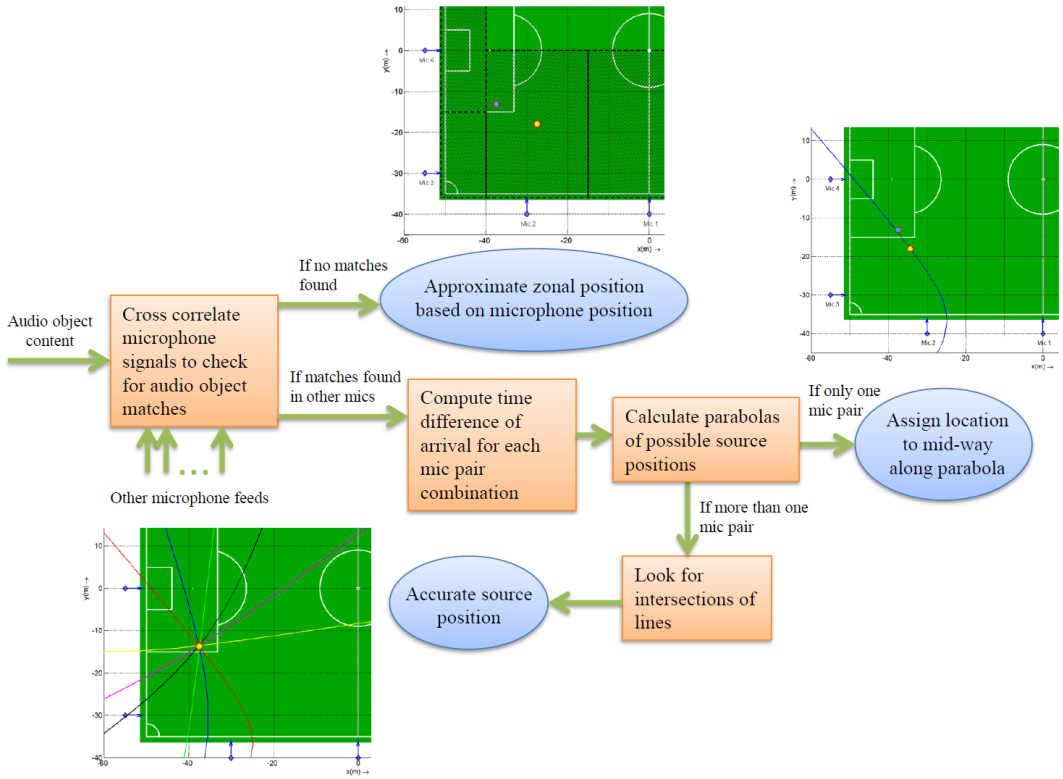

Fig. 11 Flow diagram for the possible source positioning scenarios based on the available capture data. The yellow dots show the predicted source location, the turquoise dots show the actual position

In some cases, the captured content provides insufficient information to be able to accurately identify the source position and in this case an approximation of the source position is made based on the information available. For example if only one microphone picks up the OPAE or the correlation between two microphones is not good enough to determine the relative time delay, the source must be positioned approximately based on the principle capture zone of the active microphone. Additionally, if the OPAE can only be found in two microphone signals the source can only be approximately positioned along one hyperbola, in which case it can be positioned along the hyperbola and halfway into the principle capture zone of the active microphone as shown in Fig. 11. Positioning the source half-way into the principle capture zone in this way minimizes the probability for very large position errors.

Testing the algorithm

Extraction of audio object content

To test the algorithm, a one minute section of audio was selected at random from each of the 12 pitch-side shotgun microphones. The number of ball-kicks and whistle-blows in each section was counted manually, then the audio was fed through the algorithms and the number of detected objects counted. Also counted were the number of ball-kicks that were in the original audio stream but that were missed by the algorithm and the number of false object detections for each microphone signal.

Results

The results from the content extraction experiment are shown in Table 1. The table shows the number of correct object detections in each test section of audio, additionally, it shows the number of false object detections, that is to say, when the extractor incorrectly determined a section of audio was a ball-kick or whistle-blow. The final column states the number of OPAE which were present in the original audio but were not interpreted as OPAE by the algorithm.

Microphone position	***BALL-KICKS***			***WHISTLE-BLOWS***		
	Correct detection	**False detection**	**Missed**	**Correct detection**	**False detection**	**Missed**
1. Left hand Goal	8	1	1	3	0	0
2. Far Left Corner	8	2	1	0	0	0
3. Far left wing	1	1	0	0	0	0
4. Far Centre	8	2	0	1	0	0
5. Far Right wing	11	2	0	0	1	0
6. Far Right Corner	8	5	0	0	0	0
7. Right Hand Goal	11	3	0	0	0	0
8. Near Right Corner	7	1	0	0	1	0
9. Near Right wing	7	0	0	0	0	0
10. Near Centre	9	0	2	1	0	0
11. Near Left wing	7	0	0	0	0	0
12. Near Left Corner	1	0	0	3	0	0

Table 1. Results from the ball-kick and whistle-blow extraction test. The microphone numbers correspond to the numbers on Fig. 1

Discussion

Ball-kick extraction

The results from the ball-kick extraction show that the algorithm is sufficiently robust to extract over 95% of the ball-kicks that could be discerned in the raw audio feed. There were however a small number of false detections which could potentially be problematic in a broadcast scenario. There are three possible causes of the false detections. Firstly an on-pitch noise other than a ball-kick (possibly someone on the pitch shouting for example) could be picked up if the signal happened to contain a similar transient structure to a typical ball-kick. This case is not too problematic for a broadcast scenario, as the motivation of the algorithm is to extract on-pitch sounds, so this case could be considered a success as the real sound is at least coming from the pitch even if it is not a ball-kick. Secondly, noise from the crowd could be picked up and detected as a ball-kick. This situation is difficult to avoid, especially where it is the culture for the crowd to bang drums or seats in support of their teams. This type of false detection however can be greatly reduced and even eliminated if more complex recording techniques are used as described in section 6.1 where multiple microphones at each location can be used to discern whether the sounds are coming from in front or behind the microphones and extracting, accordingly, only the events that are detected in front of the microphone, i.e. the on-pitch sounds. The third cause of false detections is the public address system. The level of this system is high and

as it is used for speech it contains lots of transient energy, in the case of this captured material the frequency equalisation was such that the content was low frequency biased, hence on one occasion the algorithm detected the content as an OPAE. This could be eliminated by either decreasing the sensitivity of the algorithm or having a link to the public address system that switch off the extractor for the duration of the announcement or by using a matched filter to reduce the level of the system in the microphone feeds.

An interesting feature of the results in Table 1 is that most of the false detections of ball-kicks occurred in the microphones situated in the corners. This is to some extent unsurprising, as those are the microphones that are closest to the crowd. As the microphones used are highly directional in this scenario they exhibit a large rear lobe, thus picking up a lot of energy from the nearby crowd which contains a lot of transient energy. For future versions of the algorithm a microphone position dependent sensitivity should be introduced such that these microphones are less sensitive to transient and only detect strong OPAE such as a corner kick.

Whistle-blow extraction

As the test data used for this experiment contained comparatively few whistle-blows, the algorithm was further tested with 12 shorter sections of audio each containing one whistle-blow each. In this case the algorithm was very robust and successfully detected all 12 whistle-blows without any false detections. The two false detections in Table 1 were caused by crowd noise, in one case, someone whistling at a frequency within the range of a referee's whistle; the other false detection was from singing in the crowd. As with the ball-kick detection, these false detections can be suppressed by utilising more advanced recording techniques such as those described in section 6.1.

The results for both the ball-kick and whistle-blow experiments highlight the importance of fine-tuning the algorithms. There is a tradeoff here between not missing any OPAE and falsely detecting them. For a broadcast scenario the threshold and coefficients would need to be optimised such that false events were not detected as it is considered more detrimental to incorrectly position an audio object on the pitch rather than to fail to pick one up. Turning the sensitivity of the algorithm down is not problematic because in most cases, if the OPAE is not clearly detected in a microphone feed, it is because the sound occurred outside the principle capture zone of that microphone and hence will be most likely detected in another microphone feed instead. To improve both this situation and the false detection of sounds from the crowd, microphones with a broader directivity could be used. This would not only increase the principle capture zone of the microphones but would also reduce the rear lobe of the microphone such that less crowd noise would be picked up. Currently highly directional microphones are used in broadcasts to reduce crowd noise from the side of the microphone; however this results in areas of the pitch which are not covered by any microphone. Using a broader directivity microphone would decrease these quiet zones and decrease crowd noise from the rear of the microphone. The additional crowd noise that this would introduce from the sides would be less problematic for the extraction algorithm because the incident crowd noise would be further away and therefore contain less transient energy that that from the rear of the microphone.

Audio object positioning

Testing the performance of the audio object positioning algorithm is fairly difficult, as the exact positions the OPAEs in a real football match are not known accurately, so there is no clear reference. Consequently a simple test case was set up on a football pitch while no match was taking place. Controlled ball-kicks and whistle-blows were made at 13 different locations on the pitch and were recorded at 4 microphone positions that would be used in a real match as shown in Fig. 12. The microphones were Sennheiser 416 shotgun microphones which are the microphones commonly used in real broadcast situations. The four microphone signals were recorded simultaneously using a 4-channel solid-state recorder, thus ensuring sample accuracy between channels. For both the ball-kicks and the whistle-blows the captured audio for each source position was fed in to the source positioning algorithm and a position estimated. This estimated position was then compared with the measured source position.

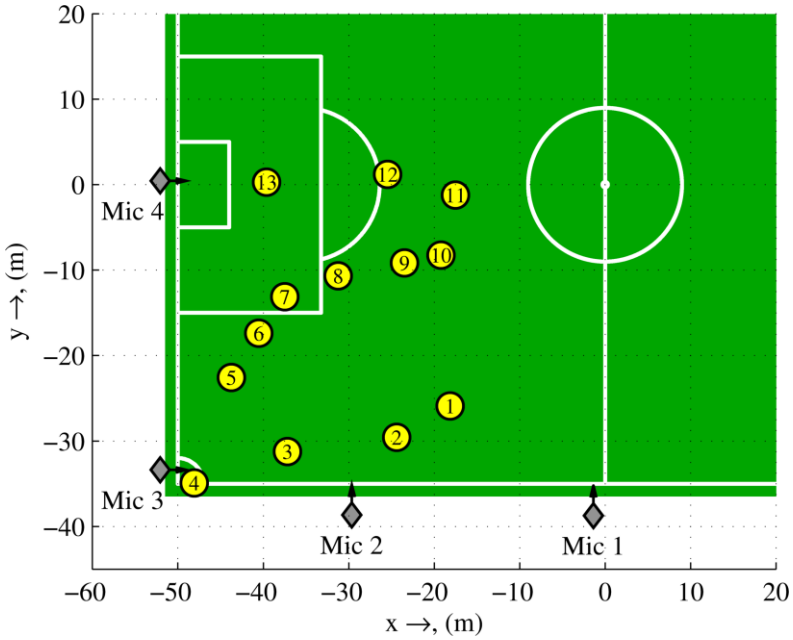

Fig. 12 Diagram of test setup for the source positioning experiment with the yellow circles showing the 13 source positions

The test was designed to test the validity of the approach to object positioning that has been employed. The test data was collected in a low noise environment allowing noise to be added to each of the microphone channels separately to determine the algorithm's ability to operate correctly in the high noise environment of a real match scenario.

Results

Once the source position had been estimated for each test location, the error between the manually measured position and the estimated position was calculated. Table 2 shows the results for both the whistle-blow and the ball-kick test sources. The table shows the error in estimated source location and also the standard deviation (STD) that gives a measure of the variance in all of the calculated points of intersection from the hyperbola pairs. Hence a larger STD means a greater difference in the location predicted using different pairs of

hyperbolae (different microphone combinations) and is a measure of uncertainty. Fig. 12 shows how the information from each of the 4 microphone pairs is used to plot a hyperbola of possible source positions to estimate the source position.

Position	***WHISTLE-BLOWS***		***BALL-KICKS***	
	Position Error (m)	**std**	**Position Error (m)**	**std**
1	0.09	0.40	0.43	0.32
2	0.32	0.29	0.45	0.36
3	1.03	1.55	0.95	1.73
4	0.31	0.86	0.34	0.51
5	0.58	0.09	0.38	0.14
6	0.48	0.14	0.05	0.18
7	0.55	0.19	0.57	0.19
8	0.18	0.24	0.23	0.11
9	0.19	0.11	0.35	0.44
10	0.62	0.60	0.64	0.80
11	0.65	0.79	1.15	0.40
12	0.40	0.79	0.72	0.37
13	0.24	0.61	0.39	0.71
Averages	0.43	0.51	0.51	0.48

Table 2 Results for the source positioning experiment

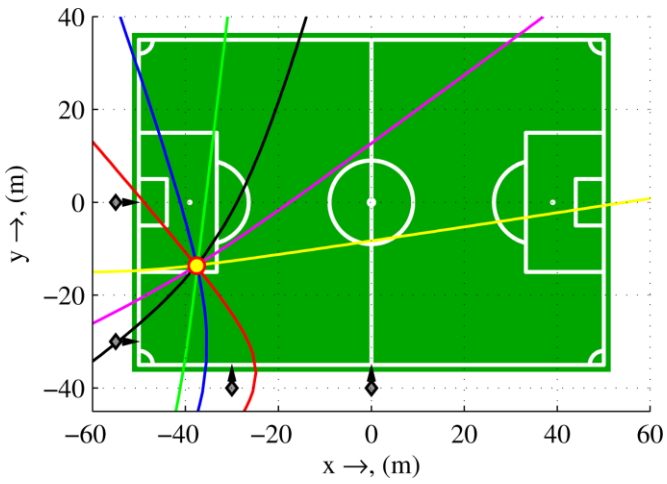

Fig. 13 Example result from source positioning experiment for a ball-kick at position 7 showing the possible source location hyperbolae from each microphone pair. The predicted source position is shown by the yellow circle.

Discussion

The results show that for the low noise case the algorithm performs very well at predicting the source positions. The average error is within the region of 0.5m which for most applications is acceptable especially as the average football pitch as dimensions of 100m × 70m. The viewer would not notice any perceptible discrepancy between the audio and video objects with this error unless they were to zoom in such that the screen only showed a very small section of the pitch,

which is an unlikely scenario. Further experiments will test the algorithm's performance in a higher noise environment. Additionally perceptual tests to prove the subjective validity of this approach are planned as a topic of a subsequent publication.

Applications

FascinatE

As mentioned in Section 1.1 the FascinatE Project is concentrating on a format agnostic approach to the broadcast of both video and the audio. For audio this means broadcasting audio objects and sound field as described in section 2.5 thus allowing the customised rendering of the audio scene at the user end whatever the format of the reproduction system. The audio object extractor presented here will be used at the production end to separate out the key audio sources from within the microphone signals at the scene and position them in space such that the they can be placed and manipulated in the sound scene at the render end. The algorithm presented here has been developed specifically for the broadcast of football but can equally be used for other sports events. For example the sound of the hits in a tennis match, the sound effects at track and field athletics etc. While the specific features of the algorithm would require adaptation to the given capture scenario, the main concepts of picking out sections from the microphones with audio salience remain. A voice activity detection module has also been written which could be used for microphones that pick up individuals' communications, but that has not been applied in this version of the audio object extractor.

Format-agnostic spatial audio reproduction

The use of audio objects allows a format agnostic rendering as audio can be mixed down into any format from wave field synthesis and higher order ambisonics to stereo. At the user end, the audio objects are positioned in the 2D or 3D sound field (depending on the user's audio system); the sound field component is decoded at the user end and then combined with the audio objects and the corresponding loudspeaker signals are then derived for the specific loudspeaker setup.

Non-FascinatE

Automatic content retrieval

For non-FascinatE applications the audio object extractor can be used for offline navigation of the originally broadcast content. Searching through the broadcast media could also be facilitated using the audio object extractor, allowing the retrieval, for example, of every time the referee blew his whistle, or the passage of play just before the goal was scored etc. As each audio object is extracted, metadata is also generated, pertaining to the type of object, when and where it appears in the audio scene and levels of confidence, i.e. how significant the object is in the scene. All of this information allows the offline navigation of the originally broadcast content such that search algorithms could pick out the most significant and interesting sections of a match by looking at, for example, all the instances where the referee blows his whistle or by analysing the crowd microphones and looking for the times when the levels are particularly high.

Assisted production and highlight generation

The audio object extraction data could also be used to assist the production of both live broadcast and highlights programming. This would involve determining which area(s) of the pitch contain the most significant action and thus selecting which cameras should be made active at that point in time. It could also assist with shot framing and selecting suitable content for a highlights program.

Automatic mixing

As described in section 1.2, football is currently dynamically mixed by the broadcast sound engineer such that each of the pitch-side microphones is only added into the mix when the play is within the vicinity of the microphone in question. The audio object extractor can be used in this scenario to automatically add in to the mix the microphones that are near the play and indeed to do so only when there is significant audio content in the microphone feed.

Further improvements

Using additional microphones

The constraints of broadcasting live sport events mean that there are a limited number of approved pitch-side microphone positions available for audio capture. This makes the extraction and positioning of audio objects difficult as the array of microphones is so sparse. However it is possible to include more than one microphone at each location without any additional problems. A second microphone can be placed at each position but with the capsule slightly behind the broadcast microphone. This second microphone is not used for the broadcast but only used for the analysis and extraction of the OPAE. With the positions of the microphones offset with respect to each other an initial analysis stage can be performed to determine whether the event is in front or behind the microphone, this can be done using a cross-correlation method to determine the relative delay between the audio arriving at each microphone capsule. This data can then be used to discard any audio objects that are from the crowd and will only allow the significant on-pitch objects to be broadcast. Further tests are planned at an English Premier League match to test the validity of this approach for a real broadcast system.

Using camera data for better source positioning

Using audio data alone for the localization and positioning of audio sources can work well but in some circumstances, the lack of audio data available from more than one microphone may make it difficult to position the source accurately on the pitch. With this is mind it is possible to use the data from the cameras to locate the position of the sources on the pitch and from this data to either turn on or point multiple capsule microphone polar patterns in the right direction for better capturing or to position the source more accurately in the sound field for rendering.

The camera data that can be used includes, camera zoom, pan/tilt position and focal length. Increasingly broadcast cameras are utilising camera heads that are able to provide this kind of metadata with respect to time. For the main broadcast camera (whose job it is to follow the play around) this gives an approximate

position of the main on-pitch action and therefore the main audio source(s). It would provide only a rough position but when combined with data from the other cameras, it is possible to perform some triangulation to get better source positions as shown in Fig. 14.

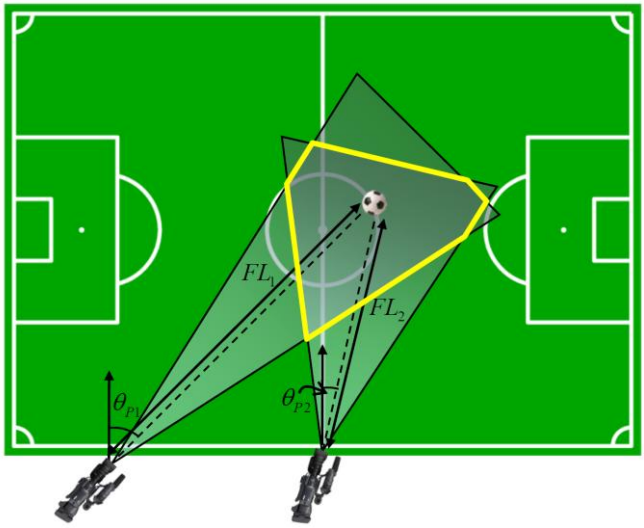

Fig. 14 Using camera data to better position audio sources

The potential of using this approach does raise the question as to whether the visual information from the video content could be used to localise the position of the ball using visual tracking systems and a artificial ball kick inserted at a corresponding time. Problems with this idea are that it would be difficult to detect when the ball was actually kicked and that each kick has a different sound depending on how hard the ball is kicked. Additionally this method would not allow the detection of a referee's whistle-blow.

Using player tracking for better source positioning

Another part of the FascinatE project is implementing a player tracking system. For FascinatE, this will allow the user to keep their view on one particular player or even to track the ball and follow the ball around the pitch automatically. One could however imagine a hybrid system that included visual tracking to help inform the audio object extraction process as to the source position once an OPAE is detected in the microphone streams. One can even imagine tracking the referee as well as the player which would enable an increased accuracy of whistle-blows.

This is a topic for further work.

Capturing more audio objects

A further extension to this work will include the capture of additional OPAE including players'/managers' communications. To this end a voice activity detection algorithm has been written and will be implemented in future versions of the system. Additionally, for a better spatialisation of the recorded scene, the crowd can be recorded using more microphones such that it can be spatialised, giving different audiences control over the level of the crowd noise they here from the home and away fans respectively.

Recording and transmitting the audio scene in the object-based manner described in this paper also allows the listener to have control over the respective levels of the commentary (foreground) and the crowd (background/ambience) which can be used as a means of improving the speech intelligibility for hearing impaired listeners (Shirley and Kendrick, 2004). It also enables the possibility of creating a hyper-real broadcast where the on-pitch (diegetic) sounds can be made louder to add to the drama of the football match an audio object approach enables the viewer to make these choices at the rendering end. Non-diegetic sounds such as the crowd could also be adjusted in level, giving a greater sense of presence in the scene.

Conclusions

A robust system has been demonstrated for extracting low level audio objects for object-based, format agnostic reproduction for football coverage as part of the EU FP7 FascinatE project. Using a standard microphone configuration as currently used for coverage of the English Premier League the system has demonstrated reliable, close to real-time, audio object extraction with a mean localisation accuracy of 0.5m under experimental conditions, and has been shown to be reliable in extracting audible ball kicks from recorded broadcast microphone feeds from coverage of an English Premier League fixture. An example audio format is presented which can combine both object-based audio and higher order ambisonic recordings in order to provide interactive and format agnostic reproduction. Future development work has been identified to further enhance the system in providing real-time automated mixing of microphone feeds and for offline navigation and search of recorded material from a live sports event.

Acknowledgments

This research project work is part of the FascinatE project which has received funding from the European Union's Seventh Framework Programme (FP7/2007-2013) under grant agreement no: 248138.

References

Andrews, P. (**2011**). “The Sound of Sport: What is real?,” IoA Reproduced Sound(Brighton).

Barsanti, R. J., and Tummala, M. (**2003**). “Wavelet-based time delay estimates for transient signals,” IEEE Conference on Signals, Systems and Computers.

Benesty., J. (**2000**). “Adaptive eigenvalue decomposition for passive acoustic source localization,” J. Acoust. Soc. Am. **107**, 384 – 391.

Bove Jr, V. M. (**1995**). “Object Oriented Television,” SMPTE Journal **104**, 803 – 807.

Bove Jr, V. M. (**1996**). “Multimedia based on object models: Some whys and hows,” IBM Systems Journal **35**, 337 – 348.

Carey, R., and Bell, G. (**1997**). *The Annotated VRML 97 Reference Manual* (Addison-Wesley Professional).

Carter, G. C. (**1993**). *Coherence and time delay estimation: an applied tutorial for research, development, test, and evaluation engineers* (IEEE Press).

Cengarle, G., Mateos, T., Olaiz, N., and Arum, P. (**2001**). "A New Technology for the Assisted Mixing of Sport Events: Application to Live Football Broadcasting," 128th Conv. Audio Eng. Soc.

Chen, S. C., Shyu, M. L., Zhang, C., Luo, L., and Chen, M. (**2003**). "Detection of soccer goal shots using joint multimedia features and classification rules In Proceedings of the (pp 36-44)," 4th International Workshop on Multimedia Data Mining (MDM/KDD2003).

Cheng, Z., and Tjhung, T. T. (**2003**). "A new time delay estimator based on ETDE," IEEE Transactions on Signal Processing **51**, 1859 – 1869.

Choi, S., Cichocki, A., Park, H.-M., and Lee, S.-Y. (**2005**). "Blind source separation and independent component analysis: A review," Neural Information Processing-Letters and Reviews **6**, 1 – 57.

Do, H., Silverman, H. F., and Yu, Y. (**2007**). "A real-time SRP-PHAT source location implementation using stochastic region contraction (SRC) on a large-aperture microphone array," IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2007.

Geier, M., Ahrens, J., and Spors, S. (**2010**). "Objectbased Audio Reproduction and the Audio Scene Description Format," Organised Sound **15**, 219 – 227.

Gerzon, M. A. (**1973**). "Periphony: With-height sound reproduction," J. Audio Eng. Soc **21**, 2 – 10.

Gerzon, M. A. (**1985**). "Ambisonics in Multichannel Broadcasting and Video," J. Audio Eng. Soc **33**, 859 – 871.

Grennberg, Anders and Sandell, M. (**1994**). "Estimation of subsample time delay differences in narrowband ultrasonic echoes using the Hilbert transform correlation," IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control **41**, 588 – 195.

Hoffmann, H., Dachselt, R., and Meissner, K. (**2003**). "An Independent Declarative 3D Audio Format on the Basis of XML," 2003 int. conf. on Auditory Display, ICAD(Boston, MA, USA).

Jakobsson, A., Swindlehurst, A. L., and Stoica, P. (**1998**). "Subspace-based estimation of time delays and Doppler shifts," IEEE Transactions on Signal Processing **46**, 2472 – 2483.

Kim, H.-G., Moreau, N., and Sikora, T. (**2006**). *MPEG-7 audio and beyond: Audio content indexing and retrieval* (Wiley).

Knapp, C., and Carter, G. (**1976**). "The generalized correlation method for estimation of time delay," IEEE Transactions on Acoustics, Speech and Signal Processing **24**, 320 – 327.

Kyriakakis, C. (**1998**). "Fundamental and technological limitations of immersive audio systems," Proceedings of the IEEE **86**, 941 – 951.

Lindsay, A. ., and Herre, J. (**2001**). "MPEG-7 and MPEG-7 Audio - An Overview," J. Audio Eng. Soc **49**, 589 – 594.

Metz, C. E. (**1978**). "Basic principles of ROC analysis," Seminars in nuclear medicine **8**, 283 – 298.

MPEG. (**1998**). *ISO 14496-3 (MPEG-4 Audio) Final Committee Draft. MPEG Document W2203*.

Peters, N. (**2008**). "SpatDIF – The Spatial Sound Description Interchange Format," 2008 International Computer Music Conference, ICMC(San Francisco, CA, USA).

Pihkala, K., and Lokki, T. (**2003**). "Extending SMIL with 3D Audio," 2003 int. conf. on Auditory Display, ICAD(Boston, MA, USA).

Scheirer, E. ., Vaananen, R., and Huopaniemi, J. (**1999**). "AudioBIFS: Describing audio scenes with the MPEG-4 multimedia standard," IEEE Transactions on Multimedia **1**, 237 – 250.

Schreer, O., Feldmann, I., Weissig, C., Kauff, P., and Schäfer, R. (**2013**). "Ultrahigh-Resolution Panoramic Imaging for Format-Agnostic Video Production," Proceedings of IEEE **101**, 99 – 114.

Shirley, B. G., and Kendrick, P. (**2004**). "ITC Clean Audio Project," 116th Conv. Audio Eng. Soc.2(Berlin).

Silverman, H. F., Yu, Y., Sachar, J. M., and Patterson III, W. R. (**2005**). "Performance of real-time source-location estimators for a large-aperture microphone array," IEEE Transactions of Speech and Audio Processing **13**, 593 – 606.

Torkkola, K. (**1999**). "Blind separation for audio signals–are we there yet?," Workshop on Independent Component Analysis and Blind Signal Separation(Aussois, France).

Vincent, E., Jafari, M. G., A., A. S., Plumbley, M. D., and Davies, M. E. (**2005**). *Blind audio source separation, Tech Report C4DM-TR-05-01*.

Wang, J., Xu, C., Chng, E., and Tian, Q. (**2004**). "Sports highlight detection from keyword sequences using hmm," IEEE International Conference on Multimedia and Expo (ICME'04)pp. 599 – 602.

Watlington, J. A., and Bove Jr, V. M. (**1997**). "A system for parallel media processing," Parallel Computing **23**, 1793 – 1809.

Westner, A. G. (**1998**). *Westner, Alexander George. Object-based audio capture: separating acoustically-mixed sounds. Diss. Massachusetts Institute of Technology, 1998.*

Zongchuang, L., Xingzhao, L., and Yongtan, L. (**2002**). "A modified time delay estimation algorithm based on higher order statistics for signal detection problems," 6th IEEE Int. Conf. on Signal Processing.