Multi-modal Visual Data Registration and Web-based Visualisation

Hansung Kim, Alun Evans, Josep Blat, and Adrian Hilton, Member, IEEE

Abstract—Recent developments of video and sensing technology can lead to large amounts of digital media data. Current media production rely on both video from the principal camera together with a wide variety of heterogeneous source of supporting data (photos, LiDAR point clouds, witness video camera, HDR and depth imagery). Registration of visual data acquired from various 2D and 3D sensing modalities is challenging because current matching and registration methods are not appropriate due to differences in formats and noise types of multi-modal data. A combined 2D/3D visualisation of this registered data allows an integrated overview of the entire dataset. For such a visualisation a web-based context presents several advantages. In this paper we propose a unified framework for registration and visualisation of this type of visual media data. A new feature description and matching method is proposed, adaptively considering local geometry, semi-global geometry and colour information in the scene for more robust registration. The resulting registered 2D/3D multi-modal visual data is too large to be downloaded and viewed directly via the web browser while maintaining an acceptable user experience. Thus, we employ hierarchical techniques for compression and restructurings to enable efficient transmission and visualisation over the web, leading to interactive visualisation as registered point clouds, 2D images, and videos in the browser, improving on the current state of the art techniques for web-based visualisation of big media data. This is the first unified 3D web-based visualisation of multi-modal visual media production datasets. The proposed pipeline is tested on big multi-modal dataset typical of film and broadcast production which are made publicly available. The proposed feature description method shows two times higher precision of feature matching and more stable registration performance than existing 3D feature descriptors.

Index Terms—Multi-modal visual data processing, 2D-3D registration, 3D feature descriptors, 3D feature matching, Progressive rendering, WebGL visualisation.

I. INTRODUCTION

The development of visual sensor technology over recent decades has led to various 2D/3D media content acquisition devices available in our lives. In digital media production, broadcasting, game design or virtual/augmented reality systems, the trend is to deal with big data captured not only from video or photography but also from a variety of digital sensors. The appearance of a scene can be captured using different digital video cameras, from 4K/6K and professional HD cameras, to those of mobile phones. Time-of-flight or Kinect-like RGBD sensors can capture video-rate depth information, while 3D laser scans create a dense and accurate geometrical point cloud of the scene. Spherical high dynamic range imaging (HDR) scanners capture full 360° texture and illumination data which is important for backplates and relighting. There may be other data sources such as video capture using drones or large collections of images captured with high-resolution DSLR cameras. There is an explosion in the volume, variety and complexity of data that outstrips the capacity of current methods to manage, analyse and visualise them. In digital production it is typical for a single film to use >1PB of storage for media assets with requirements increasing year-on-year. For example 350TB was allocated to the footage from various capture devices for the production of John Carter of Mars (2012), and Avengers: Age of Ultron (2015) is reported to have required >1PB of storage. The types of data that are typically captured using visual sensors for film production, games, VR experience and TV production is shown in Table I. While data storage is cheaper than ever, all of this data need to be sorted, indexed and processed, which is a largely manual task.

We previously presented a multiple HD video camera system for studio production [1], which addressed the registration of multiple cameras to the world coordinate through calibration for 3D video production of actor performance. This has been extended to outdoor capture by combining multiple HD cameras and a spherical camera [2]. Dynamic objects captured by HD video cameras and static background scene scanned by a spherical camera were registered to the world coordinate system. In this paper, we extend the capture system further to allow automatic registration of the wide variety of visual data capture devices typically used in production.

A key issue is automatic registration of multi-modal visual data into a common coordinate system to allow visualisation and verification of the completeness of the data. This is essential to validate data collection at the point of capture. The task of handling 3D data is not merely a case of extending the dimensionality of existing 2D image processing. Data matching and registration is more difficult because 3D data can exist in different domains with different types of formats, characteristics, density and sources of error. In this paper, we introduce a unified 3D space (Fig. 1) where 2D and 3D data are registered for efficient data management and visualisation.
2D data are registered via 3D reconstruction because direct registration of 2D to 3D structure [3], [4] is difficult to be applied for general multi-modal data registration. We assume that multiple 2D data exist for the same scene so that 3D geometric information can be extracted.

This unified space grounded in registration should be visualised integrating multiple 2D data (for example, video footage from several cameras) with raw 3D data (for example, laser-scan point clouds). A web (or browser)-based application permits seamless mixing of 2D and 3D in a single context, allowing users to more quickly understand and navigate through the scene [5]. A web application has further advantages: it is platform independent, accessible remotely and easy to update and maintain. It requires no external software to be installed, is suited for access from all over the world, and supports collaborative workflows. In this sense, there is a strong drive for many modern visualisation applications to be web-based [6]. However, it requires great care in both its design and implementation, as a poorly designed hybrid 2D-3D visual experience can be incoherent in its use, and awkward to create. On the other hand, the raw multi-modal data discussed in this paper is large (and thus difficult to transfer over the web), and by its very nature has no consistent format or structure. Web-based 3D rendering is an emerging subject which has recently reached a new level of maturity, with recognition that the challenges faced are considerably different to those of offline rendering [7]. The most relevant issues are the time taken to download the dataset to a remote client, and the challenge of visualising such big data in a (relatively underpowered) web-browser. This paper addresses directly these challenges: the combination of modalities, the efficient use of bandwidth, and the processing at the client side (which has implications on usability).

The following are the main contributions of this paper.

- A complete system from capture to visualisation through data processing and transfer for efficient management of multi-sensory visual data from 3D and 2D modalities.
- A robust multi-modal visual data registration method using a multi-domain (colour, local geometry and semi-global geometry) feature descriptor and hybrid RANSAC-based matching method.
- Comprehensive evaluation of 3D feature detectors and descriptors for registration of 3D data of the built environment from multiple visual sensors.
- A progressive, Level-of-Detail (LOD) web-based visualisation of multi-modal visual datasets for efficient data transfer and interactive rendering.
- A public multi-modal database captured with a wide variety of devices in different environments to assist further research.

### II. Related Work

#### A. Multi-modal Visual Data Registration

In general visual media processing, there has been some research for 2D/3D data matching and registration via Structure-from-Motion (SfM) and feature matching. 2D-3D registration between two modalities such as images to LIDAR [4], [8], [9], images to range sensor [10], [11] or spherical images to LIDAR [12], [13] has also been investigated. To the best of our knowledge, 2D and 3D data registration and visualisation for three or more visual modalities in general environments had not been investigated until our preliminary research. Initially, we tested existing 3D feature descriptors on multimodal registration [14] and applied them to different domains (local, keypoint and colour domains) in order to verify the influence of colour and feature geometry [15]. The work in this paper goes far beyond our previous works. We propose a full 2D/3D multi-modal data registration pipeline from capture to visualisation using multi-domain feature description and hybrid RANSAC-based registration based on the observations from our preliminary research. The proposed algorithms are tested on public multi-modal datasets, and objective analysis of feature matching and registration performance is provided in this paper.

#### B. 3D Feature Detection and Descriptors

Feature (keypoint) detection identifies the location of distinct points in terms of variation in data. There have been many 3D keypoint detectors developed and evaluated for high distinctiveness and repeatability on 3D point clouds [16], [17]. However, the majority of the best performing detectors are not suitable for multi-modal data registration because source models can have different colour histograms, or errors in their geometry, according to the characteristics of the capture device. We prefer classic detectors which produce a relatively large number of evenly distributed keypoints such as Kanade-Tomasi detector [18] used in our previous research.

Feature descriptors define the characteristics of keypoints. Restrepo and Mundy [19] tested local 3D descriptors for registering 3D point clouds reconstructed by multi-view stereo methods. Recently Guo et al. [20], [21] performed a comprehensive evaluation of local feature descriptors on various datasets from different modalities, but the test was carried out not across modalities as in this work but only within single modality in each data set. We performed similar evaluation.
WebGL and associated HTML5 APIs (such as WebAudio) are fully supported in the latest versions of all major browsers. It is now possible to access dedicated graphics processing hardware directly from the browser (via Javascript). It is difficult to balance colours between modalities. Appearance information cannot be trusted for non-Lambertian surface or repetitive patterns. The descriptors are concatenated without any priority or weight in [24] and [25], which leads to poor performance when the matching is dominated by one descriptor as demonstrated in our preliminary research [15]. In this paper, we propose a novel matching and registration algorithm adaptively considering multiple descriptors using a hybrid RANSAC technique.

C. Web-based Visualisation of Multi-modal data

Jankowski et al. [5], [26] demonstrated that a so-called "dual-mode" interface, integrating text and 3D contexts, outperforms a more classical approach where they are separated (even taking into account modality switches). The visualisation of the unified space proposed in this paper requires such hybrid integration - of 3D (more challenging than that of Jankowski), and a wealth of layered 2D data and metadata. A HTML5 web context is suitable, in this regard, as it allows (and indeed encourages) the interplay of multimedia data. 3D web pages are relatively uncommon (compared to 2D pages), and for several years were mostly represented by declarative technologies developed in the academic domain [27], [28]. However, 3D web applications have been growing in popularity since the release of WebGL in 2011. WebGL is a web-specific version of the OpenGL graphics API (more specifically of the restricted embedded systems API, OpenGL ES2.0), and allows access to dedicated graphics processing hardware directly from the browser (via Javascript). It is now fully supported in the latest versions of all major browsers. WebGL and associated HTML5 APIs (such as WebAudio\(^1\)) are in many respects enabling technologies, as they break down the barriers for the development of browser-based multimedia applications. Nevertheless, they also open up new research challenges for the best way to transmit and interact with hybrid data (be it 3D, 2D image/video, audio, or text).

3D data is typically large, and transferring it to a remote client for rendering is a persistent problem for all web 3D applications. This is particularly relevant for our work, in that the multimoal visual data is stored in files which reach many hundred of megabytes in size - simply "waiting for them to download" does not provide an optimal or satisfactory user experience. While a naive approach might be to simply compress the data using any number of established and powerful algorithms, Limper et al. [29] show that straightforward data compression may not necessarily be the solution, as the decompression time in a browser-based context may outweigh any benefits gained in terms of compressed data, particularly as bandwidth speeds increase. For a more complete overview of these issues, and the current state of the art with respect to web-based 3D, including techniques of remote rendering and progressive transmission, we refer the reader to a recent survey paper [7].

In our preliminary research in this field, we presented a similar progressive visualisation of large point cloud data, where the data is pre-processed, in an off-line step, into a hierarchical data structure [30]. Web-based rendering of very large point-clouds is tackled by [31], which uses a level-of-detail approach to ensure the number of points rendered does not saturate the browser application. Only a single point cloud visualisation in a web environment was dealt with in [30], but we present algorithms and interface for the simultaneous visualisation of multiple point clouds, intertwined with 2D image and video data in a single web-based visualisation platform in this paper. The results compare favourably for transmission times for the different but related problem of mesh visualisation in [32].

III. System Overview

Figure 2 shows the overall process for multi-modal data registration and visualisation. We use colour 3D point clouds as a common input format for 3D feature detection and matching because some inputs may not have mesh connectivity information. 3D data from 3D sensors or proxy computer graphics (CG) objects are directly registered and 2D data are registered via 3D reconstruction techniques such as stereo matching or Structure-from-Motion. In 3D reconstruction, camera poses are extracted so that the original capture lo-

\(^1\)http://www.w3.org/TR/webaudio/
cations and orientations can be simultaneously transformed in registration.

Point clouds from different modalities have different density, and some of them have irregular sample distribution even in the same scene. For example, point clouds from a LIDAR scanner or spherical images become sparser as the distance from the capture device increases. This may cause bias in feature detection and description. We apply a 3D voxel grid filter which samples vertices in a uniform 3D grid to make the density of point clouds relatively even.

Keypoints are detected by the combination of a 3D Kanade-Tomasi detector [18] and 3D SIFT detector [33] (Section V.A). Then multi-domain 3D features are extracted in local, keypoint and colour domain as a 2D vector for each keypoint (Section V.B). The extracted feature descriptors from different modalities are matched to find the optimised registration matrix to the target coordinate system (Section V.C). The point cloud registration is refined over the whole point cloud using the Iterative Closest Point (ICP) algorithm [34].

The complete dataset is then organised and processed into a representation suitable for transmission over the web. Video files are compressed using the OGG/Theora codec, and thumbnails are created from all image and video files. 3D point cloud data is entered into an octree data structure, which is traversed breadth first to create a series of binary files, ready for progressive download to the client. The final visualisation is a web application engine which mixes both 2D video, 2D image and 3D WebGL contexts to allow users to navigate through the scene in an interactive manner. The application is designed to work on handheld devices as well.

IV. INPUT MODALITIES

We consider a wide range of 2D/3D and active/passive sensors commonly used in various fields.

A. Light Detection and Ranging (LIDAR) sensor

LIDAR is an active sensing device using a light pulse signal to acquire 3D scene geometry. It is one of the most accurate depth ranging devices but has the limitation that it retrieves only a point cloud set without colour or connectivity. However, some recent LIDAR devices provide coloured 3D structure by mapping photos simultaneously taken during the scan. We have verified that colour information is useful in multimodal data registration in our previous work [15], so we use FARO Focus 3D X130\(^2\) to obtain coloured 3D point clouds in this work. Multiple scans acquired from different viewpoints are manually registered and merged into a complete scene structure using markers in the scene and the software tool provided with FARO. We do not use our automatic registration method for this partial scan registration because this LIDAR model will be used as a ground-truth target reference in our evaluation.

B. Spherical Imaging

A spherical camera captures a full surrounding scene visible from the camera location. Omni directional imaging is useful for environmental texture map generation or lighting source detection, but it always requires post-processing to map the image in spherical coordinates to other images captured in a different coordinate system [35]. We assume that the scene is captured as vertical stereo pairs to allow dense reconstruction of the surrounding scene for automatic registration. We use Spheron\(^3\), a spherical line scan camera and follow the stereo matching and reconstruction approach in [36].

C. Photographs

Digital photographs are the most common source of scene information. 3D reconstruction and camera pose estimation from multi-view images has been actively researched for a long time. A set of photographs can be registered to a 3D space by registering the reconstructed 3D model because the camera poses are computed during the reconstruction process. Bundler [37] followed by PMVS [38] provide a dense 3D reconstruction with camera pose estimation from multiple photos. Autodesk\(^4\) also provides an on-line image-based 3D reconstruction tool, RECAP360\(^5\). Both tools are used in our experiment.

D. 2D Videos

If a single moving video camera is used, the same approach in Section IV.C is used because video frames from a moving camera can be considered as multi-view images. In case of multiple wide-baseline witness cameras, it is difficult to get the scene geometry for automatic registration if the camera viewpoints do not have sufficient overlap. In this paper, we define 2D videos as wide-baseline fixed witness cameras capturing a common space. Camera poses are estimated by wand-based calibration [39] aligned to the origin of the LIDAR sensor.

E. RGBD Video

Consumer level low-cost RGB+Depth cameras are becoming increasingly popular. Though infra-red (IR) interference limits their validity in outdoor environments, they are still useful in indoor or shaded outdoor areas. KinectFusion [40] reconstructs a voxel volume from an RGBD video sequence by camera pose estimation and tracking. We use the Xtion PRO camera\(^6\) to acquire a RGBD video stream of the scene.

---

\(^1\)FARO, http://www.faro.com
\(^2\)FARO Focus 3D X130\(\), http://www.faro.com
\(^3\)Spheron, http://spheron.com
\(^4\)RECAP360, http://recap360.autodesk.com
\(^5\)XtionPRO, http://www.asus.com/Multimedia
F. Proxy Model

Proxy model means a simple computer graphics object which represents or symbolises real 3D objects. Proxy models are used in areas such as augmented reality, pre-visualisation, virtual maps and urban planning. They are normally generated by computer graphics, but there are some semi-automated algorithms such as plane-block-based scene reconstruction from images [41], [42]. SketchUp\textsuperscript{6} provides a semi-automatic reconstruction using vanishing points alignment. It is useful to build simple scenes but takes a long time for complex scenes. We use an axis-aligned plane-based scene reconstruction from spherical images [43] in the experiments. In feature detection and description, the plane structure is densely sampled to extract sufficient points for feature computation.

V. Multi-modal Data Registration

A. 3D Feature Detector

Keypoint detection is an essential step prior to matching and registration. There are many 3D feature detection methods developed and evaluated [16], [17]. However all detectors were evaluated for accurate 3D models generated by computer graphics or single-modal sensors. Highly-ranked detectors in those evaluations do not guarantee such high repeatability and distinctiveness for multi-modal data sets which have potentially different types of errors, sampling characteristics and distortions. For example, Heat Kernel Signature (HKS) detector [44] shows good repeatability and distinctiveness in those evaluations, but is too selective to yield a sufficient number of repeatable keypoints between cross-modalities due to geometrical errors induced from incomplete 3D reconstruction methods. A feature detector which produces a relatively large number of evenly distributed keypoints is preferred for robust multi-modal data registration. We consider colour as well as geometry to extract the most information from input datasets with outliers and different sampling resolutions. We use the combination of 3D Kanade-Tomasi detector and 3D SIFT feature detector.

The original 2D Kanade-Tomasi detector [18] uses an eigenvalue decomposition of the covariance matrix of the image gradients. In the 3D version of the Kanade-Tomasi detector, 3D surface normal vectors calculated in the volume radius of \( r_s \) are used as input. Eigenvalues represent the principal surface directions and the ratios of eigenvalues are used to detect 3D corners in the point cloud.

The SIFT feature detector [33] uses a Difference-of-Gaussian filter to select scale-space extrema then refines the results by Hessian eigenvalue test to eliminate low contrast points and edge points. We use 3D versions of the Kanade-Tomasi detector and the SIFT detector implemented in the open source Point Cloud Library\textsuperscript{7}. Parameters for 3D SIFT feature detector are defined as [Minimum scale \( S_m \), Number of octaves \( S_o \), Number of scales \( S_s \)].

B. Multi-domain Feature Descriptor

Most 3D feature descriptors rely only on local geometric or colour features. However, these descriptors are not suitable for multi-modal data registration because input sources may have a high level of geometric reconstruction error or different colour histograms. Our preliminary research [15] found that the combination of descriptors applied on different domains such as colour and geometry can improve the matching and registration performance for multi-modal data.

We use the FPFH descriptor as a base descriptor because it shows fast and stable performance in our preliminary research [14]. FPFH uses a cumulation of Simplified Point Feature Histogram (SPFH) [22]. SPFH extracts a set of tuples \([\alpha, \varphi, \theta]\) from a keypoint \( p \) and its neighbouring local points \({p_k}\), where \( \alpha \) is angle to the second axis, \( \varphi \) is an angle to the first axis, and \( \theta \) is a rotation on the \( U_W \) plane. For neighbouring local points, their \( k \)-nearest neighbours \((k-NN)\) are determined and the FPFH histogram is computed by weighted sum of their neighbouring SPFH values as Eq. (1). The weight \( \omega_k \) is a distance between points \( p \) and \( p_k \). The number of bins is set as 11 for each \( \alpha, \varphi, \theta \). Therefore one FPFH descriptor can be represented as a vector with 33 bins.

\[
FPFH(p) = SPFH(p) + \frac{1}{k} \sum_{k=1}^{k} \frac{1}{\omega_k} \cdot SPFH(p_k)
\] (1)

In this research, the FPFH descriptor is extended to multiple domains in order to utilise geometry and colour information together. For the same input point cloud with detected keypoints, three different FPFH descriptors are calculated in three different domains: Local, Keypoint and Colour. The result is represented as a 2D vector with \(33 \times 3\) bins.

FPFH in the local domain \( F_L \) defines the characteristic of local geometry calculated from a keypoint and its neighbouring local 3D points in the volume radius of \( r_l \) as normal local descriptors. FPFH in the keypoint domain \( F_K \) defines the spatial distribution of detected keypoints, which represents semi-global geometric feature of the scene. \( F_K \) is calculated from a keypoint and its neighbouring keypoints in the volume radius of \( r_k \), which is much larger than \( r_l \). Finally, FPFH in the colour domain \( F_C \) defines the colour characteristics of a keypoint and its neighbouring local 3D points in the same volume radius of \( r_l \) as \( F_L \). \( F_C \) is calculated in the same way but uses colour components instead of surface normal components. We use the CIELab colour space which is more perceptually uniform than the RGB space as proved in [25].

C. Hybrid Feature Matching and Registration

We propose the Hybrid RANSAC registration method to find an optimal 3D rigid transform matrix between feature sets. This extends the SAC-1A algorithm [22] by introducing a new distance measure with weighted sum of multi-domain FPFH descriptors. Figure 3 presents a block diagram of the proposed feature matching and registration method for the registration of keypoint set \( P \) in the source model to keypoint set \( Q \) in the target model.

The contribution of description domains in matching are adaptively selected according to the distinctiveness of the...
structure and organisation - real world big data is messy.

make it difficult to enforce strict rules on file and directory

steps, and the rendering approaches used, are discussed below.

and rendering on, a remote web client. These pre-processing

pre-processing in order to ensure its suitability for transfer to,

input data, while registered to a common 3D space, requires

approach, mixing video, image, text, and 3D displays. The

Hybrid RANSAC registration

Algorithm 1

\[ \lambda = D(p, p_{NN[1]})/D(p, p_{NN[0]}) \] (2)

The total matching cost \( D_T(p, q) \) for a source keypoint \( p \)
to a target keypoint \( q \) with multiple domains is calculated by the
weighted sum of individual domain descriptors as Eq. (3).

\[ D_T(p, q) = \lambda_D D_L(p, q) + \lambda_K D_K(p, q) + \lambda_C D_C(p, q) \] (3)

Algorithm 1 shows the registration process in detail.

A. Progressive point cloud Rendering

The registered point clouds generated by the various input
modalities presented in Section IV (from raw point cloud
scans or reconstructed image data) are initially in OFF format,
encoding position and colour of each point. File sizes range
from tens to hundreds of megabytes. Rendering such data
in an offline context is trivial; doing so in a web browser
context, however, presents two principal challenges. The first
is the simple time taken to download such data. Even with a
fast internet connection, and the coloured points represented
in binary format, and compressed using the HTTP standard gzip
gzip algorithm, it would take several seconds or even minutes
to download the data before it can be rendered. Secondly,
such a large number of points can easily overwhelm the
browser application - our initial tests, on modern hardware,
with a very simple WebGL point cloud rendering application
showed that a maximum of 3.5M points can be rendered
before the application crashes (in comparison, a similar off-
line application can render many more points).

Using a hierarchical data structure to store and transmit
the data solves both of these issues. Not only does it permit
lower-resolution versions of the dataset to be transmitted and
rendered immediately, while further data is downloaded, but
it also permits rendering of larger datasets which would not
be possible to render at full resolution in the browser. Thus,
we pre-process our data in a similar way to [30], organising
the data hierarchically into a memory efficient octree, where
the center of each node is stored, along with the mean colour
of all the points stored within it and any of its child nodes.
An offline process parses this octree breadth first and outputs
the position and colour information in a simple binary format,
which is then stored in a sequence of files. Each file contains
a maximum of 5000 entries, each entry corresponding either
to point representing a node of the octree, or a point of the
final dataset.

The browser application features as its base context a
WebGL rendering engine which downloads sequentially the
file sequence described above. Each file is processed and the

VI. WEB-BASED VISUALISATION

The web-based visualisation is based on a hybrid 2D-3D
approach, mixing video, image, text, and 3D displays. The
input data, while registered to a common 3D space, requires
pre-processing in order to ensure its suitability for transfer to,
and rendering on, a remote web client. These pre-processing
steps, and the rendering approaches used, are discussed below.

The large amount of data with which we are dealing can make it difficult to enforce strict rules on file and directory
structure and organisation - real world big data is messy.

Instead, we use a simple JSON file to store a scene description
with relative paths to the location of the relevant data. The
data is stored on a Linux-based machine running a custom
Apache2 web-service which is configured to enable HTTP
gzip compression for all the file formats which are served
the client (including the custom binary formats as described
below). This enabling of gzip for all files provides a final
compression step which is extremely fast, as it relies on a
well-understood algorithm encoded at a low level in both the
server and client-browser application, and thus adds very little
processing overhead for potentially significant reductions in
file size [30].

On the client side, the hybrid 2D-3D renderer is setup with
a base WebGL 3D context running in a HTML5 canvas ele-
ment, which is supplemented by various 2D Document Object
Model (DOM) elements, described below. Interactive scene
navigation is controlled by rotating, panning and zooming with
standard mouse/touch gestures.
data for the points uploaded to the GPU. The resolution level (i.e., depth into the octree) is tracked, so that when higher resolution data is downloaded and displayed, lower resolution data is discarded (to avoid occlusion issues). The result is that, upon loading of the web-page, an initial low-resolution version of the point cloud is quickly displayed on the screen, which is then refined to a higher resolution version as more data is downloaded, until the final point-cloud is displayed.

Multiple point clouds can be downloaded and rendered simultaneously, and hidden/shown using a simple GUI element. This capability for visualisation of multiple point clouds allows the user to quickly see the similarities and differences between the data obtained from the different modalities, which plays an important role in assessing the data quality, completeness and key requirements.

\section*{B. Video Data and Timeline}

The raw video footage recorded from the witness cameras (Section IV.D) is initially stored in uncompressed format. For transfer to the remote client it is compressed and reduced in resolution using the OGG-Theora codec at medium quality. A thumbnail image of a fixed frame from the first seconds of each video is also created. Upon loading the web-page, all videos are pre-loaded into the page DOM as HTML5 video elements, which are hidden from view using CSS (the elements are required to stream the video data from the server, but the actual frames will be rendered in WebGL as described below).

Witness cameras are represented in the 3D scene by simple plane meshes whose positions and orientations match those extracted as above. The video footage from each camera is then rendered in the 3D context, extracting the image data from the HTML5 video element and passing it as a WebGL texture, which is displayed on the relevant plane mesh for each camera (Fig. 14). This extraction of video frames from the HTML5 video element for use as textures within a 3D context is one of the major benefits of developing a hybrid interface within a web-based context, as such a pipeline in a standard desktop OpenGL context requires a greater level of software engineering and pre-processing [45].

To control playback, position and scrubbing, a simple timeline interface is drawn in a 2D canvas (Fig. 15). The timeline allows selection of which video to play, along with playback controls and a draggable timeline bar to control scrubbing. Video buffering is used to ensure that enough video data has been downloaded to pass as texture information to the WebGL renderer, and also to ensure the scrubbing interface is synchronised to the video footage. Upon selecting a witness camera in the timeline interface, the camera position in the 3D scene is instantly moved to a position just behind the plane mesh representing that camera, allowing the video to be seen within the 3D context. For performance reasons, only one video can be played at a time (the video which is selected in the timeline interface).

\section*{C. Sensor Raw Data Billboards}

The registration process described in Section V also outputs the positions of the various sensors (LIDAR, Spheron, RGBD camera, regular photo cameras, etc.), which are registered to the combined LIDAR scan for reference. To visualise these sensor positions, we render a simple mesh plane at the 3D position of the sensor within the scene, and pass a thumbnail image of the original sensor image as a texture for that plane. Unlike the similar setup for witness cameras, for the sensors we strip all rotational information out of the Model-View-Projection matrix immediately prior to rendering. This means that the plane meshes act as billboards, constantly rotating to face the camera, to best show the original sensor data. When the user clicks (or touches) the screen, a ray is fired into the scene and a simple collision detection algorithm determines whether the user has clicked on a billboard or not. If so, the 3D context is faded into the background and the original, full-resolution image of the sensor is shown in a HTML/CSS lightbox (Fig. 16).

The billboards can occasionally be difficult to spot among the rest of the point cloud data, so we have added a feature where the user can enable an interface overlay which draws coloured lines above each billboard, thus highlighting the locations of all the sensors. Different sensor types can be assigned different colours.

\section*{D. Annotation component}

One of the potential industrial benefits of the system presented in this paper is that it permits various professional users to view and interact with the same data, at the same time, while potentially being in different physical locations. The rise of remote collaborative working, seen most strongly with the popularity of online tools such as Google Docs and Dropbox, has yet to reach the 3D production and post-production world, largely due to problems which the work in this paper strives to overcome.

While a full collaborative work application lies as a potential future goal, we have implemented an annotation component, which permits users to annotate areas of the dataset, raising the possibility of those annotations being stored on a server for viewing by other users. Annotation of point cloud data is slightly more troublesome than when dealing with mesh data. In the latter case, a simple raycast-mesh collision detection is enough to detect the 3D point where the user has clicked (or tapped) on the scene. GL points however, are drawn as pixels and do not have any representative volume, thus a simple raycasting method is not sufficient. To counter this problem, we recreate in the browser context the octree used for the initial data partitioning, and calculate ray collisions on the nodes of the octree. This permits us to effectively discover the 3D point in the scene with which the user has interacted, and allows us to associate (and draw) an annotation at that point (Fig. 17).

\section*{VII. PUBLIC MULTI-MODAL DATABASE}

To support research into multi-modal data processing, we present a big multi-modal database acquired in various indoor and outdoor environments, available at: http://cvssp.org/impart/
IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY

The database includes raw capture data and 3D reconstructions for various indoor/outdoor static scenes and multiple synchronised video captures for dynamic actions in the scene. Various capture devices such as grey/colour LIDAR scanners, spherical camera, DSLR/compact still cameras, HD (1920×1080) video cameras, HD 2.7K/4K cameras and RGBD cameras were used. The HD video cameras were genlock synchronised and calibrated. The repository contains detailed notes on the capture, and some pre-processing is available to make the dataset more useful to researchers. Details can be found in the capture notes provided on the repository [46].

The proposed registration and visualisation pipeline is tested on three datasets from this repository: Studio, Patio and Cathedral. The Studio set is an indoor scene with stable lighting condition provided by KinoFlo fluorescent lights on the ceiling. The Patio set is an outdoor scene covering around 15m×10m area. The main capture area is surrounded by walls, has a symmetric structure and includes repetitive geometry and texture patterns from bricks and windows. RGBD data can be acquired for this scene without IR interference because it is shaded area. Fifteen HD video cameras were used to record main actions in the scene. The Cathedral set is a large outdoor scene covering around 30m×20m open area. The scene was captured under the direct sun light which resulted in changing brightness and shadows. Main actions were recorded by eight HD video cameras. Figure 4 shows examples of static and dynamic captures for the test scenes. As mentioned in Section IV.D, multiple HD video cameras are registered using their extrinsic camera parameters in our experiments because the cameras are too sparsely placed (little overlap) to recover the background geometry from dynamic videos.

VIII. Experiments

In order to evaluate general performance of the proposed multi-domain feature descriptor and hybrid-registration to single modality cases, we tested them on the RGB-D Scenes Dataset from University of Washington. It provides 3D colour point clouds of four indoor scenes. Each scene has 3-4 takes with different main objects and coverage for the same background scene. We randomly merged the first takes of each scene into one model as shown in Fig. 5 (a), and tried to register the second takes of each scene in Fig. 5 (b) to the merged target scene of Fig. 5 (a). Different objects and coverage of the second takes can be considered as noise or errors against the target scene, which makes the test more challenging. For objective evaluation, we generated a ground-truth registration by manual 4-points matching and ICP refinement using MeshLab.

In the experiments on the multi-modal datasets introduced in Section VII, the LIDAR scan in each scene is set as the target reference and all other models are registered to the LIDAR coordinate system. Table II shows the datasets used in the experiments. “Spherical-P” is a partial spherical reconstruction to verify the performance of part registration to the whole scene. 3D models are reconstructed for the real world scale using the reconstruction method introduced in Section IV. In reconstruction from photographs, Autodesk RECAP360 is used for the Studio and Patio scenes, and the Bundler + PMVS for the Cathedral scene to test various algorithms. HD videos are not tested for reconstruction and registration because they have been calibrated to the LIDAR coordinate system using the camera calibration process. The 3D point clouds reconstructed from 2D data for the experiments are illustrated in Fig. 6.

Ground-truth registration was generated as the same manner as the Washington dataset. Figure 7 illustrates the original datasets, ground-truth registration results and the registration error maps. The error map shows Hausdorff distance to the LIDAR model mapped in the range of 0-3m to a Blue-Red colour range. We observe that even the ground-truth registration has errors against the target model because the source model has reconstruction errors, different coverage and density. Therefore, we measure the RMS error to the ground-

truth registration points instead of the distance to the LIDAR model for the registration evaluation.

In 3D point cloud registration, the ICP algorithm requires an initial alignment. It fails in registration if the initial position is not close enough to the final position. Therefore we judge the performance of initial registration by success or failure of the following ICP refinement. We found that the ICP converges successfully if the initial registration is within 1-2m of RMS error range to the ground truth registration.

A. 3D Feature Detector

In this experiment, we evaluate existing 3D feature detectors, then analyse their influence to the registration performance. We test three feature detectors and their combinations: 3D Noble [47], 3D SIFT, 3D Tomasi, 3D Noble+SIFT and 3D Tomasi+SIFT. We do not test the combination of 3D Noble and Tomasi because both are geometry based detectors. Testing is performed on our multi-modal dataset. The range parameter \( r_s \) for surface normal calculation is set as 0.5m and 0.2m for the outdoors scenes and indoor scene, respectively. The scale parameters for the SIFT detector are set as \([S_m, S_o, S_i] = [r_t, 8, 10]\) as suggested in the original implementation.

Detected keypoints for the spherical reconstruction of the Cathedral scene are shown in Fig. 8. The Noble detector detected 4 times more points than other detectors but they are concentrated in specific regions. The SIFT and Tomasi detectors detected similar number of feature points but the result of Tomasi is more evenly spread.

The registration result using the detected keypoints in Table III clearly shows the influence of the feature detectors to matching and registration. In feature description and matching, we used the local FPFH descriptor with the parameter set \([r_t, R_{\min}, R_{\max}, I_{\max}] = [0.8\,(\text{outdoor}), 0\,(\text{indoor}), 0.2, 0.8, 8000]\) in an intuitive way considering the scale of the scenes.
Fig. 7. Ground-truth registration

No. cases in initial registration and bold ones show the scale is unknown. In Table III, figures coloured in red show been used only for the Washington datasets because their exteriors such as the Cathedral. Different parameters have range from small scale indoor scenes to large scale building not sensitive to the scene scale or characteristics across the datasets because they are fixed for all muti-modal datasets because they are not sensitive to the scene scale or characteristics across the range from small scale indoor scenes to large scale building exteriors such as the Cathedral. Different parameters have been used only for the Washington datasets because their scale is unknown. In Table III, figures coloured in red show failed cases in initial registration and bold ones show the best. No. Suc. means the number of models succeeded in initial registration for ICP, and A.RMSE means the average RMS registration error of the successful registrations. The Noble detector shows the worst performance in the single detector test in spite of the largest number of feature points because the points gathered in specific areas do not contribute to efficient matching and registration. The Tomasi detector shows the best performance among the single detectors with the largest number of successful registrations and the lowest RMS registration error. The combinations of geometric and colour detectors show better results as expected. Especially the Tomasi+SIFT detector shows good registration performance even with a normal FPFH descriptor though it still fails with the Patio set due to its repetitive geometry and texture. We use this Tomasi+SIFT detector for multi-domain feature description and Hybrid matching in the next section.

B. Feature Matching and Registration

3D feature descriptors are computed for the keypoints extracted by the combination of Tomasi and SIFT in Section VIII. A. We compared the registration performance of the proposed multi-domain FPFH descriptor and Hybrid RANSAC registration (denoted as \( F_{HYB} \)) with those of normal FPFH (\( F \)), SHOT (S), and cascade combinations of FPFH descriptors in different domains (\( F_{LK} \), \( F_{LC} \) and \( F_{LK} \)). We use the same parameter set of Section VIII. A for the multi-modal datasets and \( [r_f, r_k, R_{min}, R_{max}, I_{max}] = [0.2, 1.0, 0.05, 1.0, 5000] \) for the Washington datasets.

For matching performance evaluation, best matching pairs of all detected keypoints to the target reference are calculated and compared with the ground-truth feature matching pairs. Ground-truth feature matching pairs are defined by the closed keypoints of the target reference in the range of \( r_{gt} \) from the source keypoints transformed by the ground truth registration. \( r_{gt} \) was set as 0.03 for the Washington dataset (the scale of the 3D coordinate is unknown) and 5cm for the multi-modal dataset. As tested in [21], Precision values are computed as follows:

\[
\text{Precision} = \frac{\text{Number of correct matches}}{\text{Number of matches}} \tag{4}
\]

1) Test on single-modal dataset: Table IV shows matching precision and registration results of the Washington RGB-D scenes dataset according to the description methods. Only precision results are given here because the outlier ratio is more important in RANSAC-based registration. Avg. in the last row means the average of the whole precision values in the precision columns and the average RMS registration error of the “successful registrations” in the registration columns. In the feature matching evaluation, combination of features from various domain shows higher precision rate. Especially it shows better results both in feature matching and registration when the colour information was involved because their

<table>
<thead>
<tr>
<th>Data set</th>
<th>Noble</th>
<th>SIFT</th>
<th>Tomasi</th>
<th>N+S</th>
<th>T+S</th>
</tr>
</thead>
<tbody>
<tr>
<td>Studio-P</td>
<td>1.99</td>
<td>1.10</td>
<td><strong>0.42</strong></td>
<td>1.21</td>
<td>1.11</td>
</tr>
<tr>
<td>Studio-S</td>
<td>5.00</td>
<td>4.58</td>
<td>3.25</td>
<td>1.03</td>
<td>4.21</td>
</tr>
<tr>
<td>Studio-R</td>
<td>1.90</td>
<td>1.44</td>
<td>0.45</td>
<td>2.92</td>
<td><strong>0.28</strong></td>
</tr>
<tr>
<td>Patio-P1</td>
<td>10.41</td>
<td>15.96</td>
<td><strong>1.34</strong></td>
<td>1.71</td>
<td>1.44</td>
</tr>
<tr>
<td>Patio-P2</td>
<td>9.22</td>
<td>10.31</td>
<td><strong>1.84</strong></td>
<td>7.22</td>
<td>4.99</td>
</tr>
<tr>
<td>Patio-S</td>
<td><strong>1.13</strong></td>
<td>1.20</td>
<td>12.67</td>
<td>1.52</td>
<td>2.59</td>
</tr>
<tr>
<td>Patio-R</td>
<td>10.40</td>
<td>18.97</td>
<td>10.45</td>
<td>11.13</td>
<td>10.44</td>
</tr>
<tr>
<td>Cath-P1</td>
<td>1.69</td>
<td>1.66</td>
<td>0.61</td>
<td>1.24</td>
<td><strong>0.59</strong></td>
</tr>
<tr>
<td>Cath-P2</td>
<td>26.67</td>
<td>20.44</td>
<td>10.94</td>
<td>26.31</td>
<td><strong>0.32</strong></td>
</tr>
<tr>
<td>Cath-S</td>
<td>17.79</td>
<td>3.25</td>
<td><strong>1.26</strong></td>
<td>1.85</td>
<td>1.73</td>
</tr>
<tr>
<td>Cath-SP</td>
<td>13.45</td>
<td>13.42</td>
<td>1.63</td>
<td>1.06</td>
<td><strong>0.69</strong></td>
</tr>
<tr>
<td>Cath-PR</td>
<td>16.19</td>
<td>1.53</td>
<td>18.26</td>
<td>3.79</td>
<td><strong>0.89</strong></td>
</tr>
<tr>
<td>No. Suc.</td>
<td>4</td>
<td>5</td>
<td>7</td>
<td>7</td>
<td>8</td>
</tr>
<tr>
<td>A.RMSE</td>
<td>1.68</td>
<td>1.39</td>
<td>1.08</td>
<td>1.38</td>
<td><strong>0.88</strong></td>
</tr>
</tbody>
</table>
advantageous considering its computational complexity. Hybird RANSAC registration shows competitive performances. The proposed multi-domain feature description and colour components dominate the matching over the semi-global geometric component. The proposed hybrid matching method shows higher precision compared with other descriptors. The best 20 keypoints matches for the Patio set and 200 matches for the Cathedral set using conventional SHOT and FPFH local descriptors and the proposed multi-domain hybrid matching are visualised. The local descriptor matching results are scattered over the scene while the proposed method shows more consistent matching to the correct position.

In Table V, the Studio set shows better performance than Patio and Cathedral sets in matching and registration, and especially the colour information improves the performance of feature matching because the Studio set was captured in stable lighting condition. However, it shows poor result with the spherical reconstruction, because the Studio-S model was reconstructed from only one pair of spherical images and has large self-occlusion areas in the geometry.

The Patio scene models have repetitive structures with similar colours such as bricks and window frames. It causes relatively low feature matching rates compared with other datasets. In the registration results, we observe that some structures are mis-registered by 180° as shown in Fig. 10 (a). Keypoint descriptions \( F_K \) which considers feature distribution over a large area achieves better performance than local colour or shape descriptors due to repetitive local geometry and appearance.

In the Cathedral scene models, the appearance information is less trusted because it changes according to the capture device, capture location (direction) and time in the open outdoor environment. As shown in Fig. 10 (b), the left wing of the building is mapped to the right wing in the LIDAR model. It happens with \( F, F_{LC} \) and \( F_{LKC} \) descriptors whose local and colour components dominate the matching over the semi-global geometric component. The proposed hybrid matching and registration sorts out this bias problem. However, the colour information is more helpful than others in the case of proxy model (Cath-PR) whose distinctiveness of geometrical features are very low. SHOT descriptor also shows poor result in feature matching. This results from from the failure of defining local reference frame for SHOT descriptor.

The cascade combinations of descriptors generally show slightly better performances than the single local descriptors, but it sometimes makes worse as seen in the case of Patio-P1 with \( F_{LC} \). Cath-P2 with \( F_{LK} \), Cath-SP with \( F_{LC} \) and Cath-PR with \( F_{LK} \). They show poor performances because the features from different domains compete each other without considering their reliabilities. The proposed matching and registration method FPFH\(_{HYB} \) successfully registered all 12 datasets with high precision feature matching and low RMS registration error.

C. Web-based Visualisation

Figures 11-17 show screenshots of the various components of the visualisation. Table VI contains results showing the total time taken for the point cloud data (from all sources) to download and render, at clamped bandwith of 8Mbps. The purpose of this table is to highlight the advantage of the level-

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Precision of Feature Matching, (%)</th>
<th>Registration Error, (RMSE)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>( F )</td>
<td>( S )</td>
</tr>
<tr>
<td>Scene1-T2</td>
<td>9.56</td>
<td>13.02</td>
</tr>
<tr>
<td>Scene2-T2</td>
<td>14.47</td>
<td>15.85</td>
</tr>
<tr>
<td>Scene3-T2</td>
<td>8.60</td>
<td>12.20</td>
</tr>
<tr>
<td>Scene4-T2</td>
<td>13.55</td>
<td>7.96</td>
</tr>
<tr>
<td>No. Suc.</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>Avg.</td>
<td>11.55</td>
<td>12.26</td>
</tr>
</tbody>
</table>
TABLE V
Matching and registration results for multi-modal dataset

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Precision of Feature Matching, (%)</th>
<th>Registration Error, (RMSE)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>F</td>
<td>S</td>
</tr>
<tr>
<td>Studio-P</td>
<td>5.26</td>
<td>1.85</td>
</tr>
<tr>
<td>Studio-S</td>
<td>1.15</td>
<td>1.05</td>
</tr>
<tr>
<td>Studio-R</td>
<td>3.58</td>
<td>5.05</td>
</tr>
<tr>
<td>Studio Avg.</td>
<td>3.33</td>
<td>2.65</td>
</tr>
<tr>
<td>Patio-P1</td>
<td>1.14</td>
<td>0.93</td>
</tr>
<tr>
<td>Patio-P2</td>
<td>1.69</td>
<td>0.92</td>
</tr>
<tr>
<td>Patio-S</td>
<td>0.38</td>
<td>0.25</td>
</tr>
<tr>
<td>Patio-R</td>
<td>0.52</td>
<td>0.52</td>
</tr>
<tr>
<td>Patio Avg.</td>
<td>0.93</td>
<td>0.66</td>
</tr>
<tr>
<td>Cath-P1</td>
<td>1.72</td>
<td>1.74</td>
</tr>
<tr>
<td>Cath-P2</td>
<td>4.25</td>
<td>4.15</td>
</tr>
<tr>
<td>Cath-S</td>
<td>2.58</td>
<td>2.29</td>
</tr>
<tr>
<td>Cath-SP</td>
<td>2.24</td>
<td>1.17</td>
</tr>
<tr>
<td>Cath-PR</td>
<td>1.18</td>
<td>0.10</td>
</tr>
<tr>
<td>Cath Avg.</td>
<td>2.39</td>
<td>1.89</td>
</tr>
</tbody>
</table>

TABLE VI
Time taken (in msecs) to download and render different point clouds at three resolution levels. (First View: Initial Render of the Low Resolution Data; 50% and 100%; Percentage (Number of Points) of the Entire Dataset Rendered. The Three Scenes Were Downloaded Simultaneously. Bandwidth is Clamped to 8Mbps.)

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Cathedral</th>
<th>Patio</th>
<th>Studio</th>
</tr>
</thead>
<tbody>
<tr>
<td>Num. points</td>
<td>75838</td>
<td>276113</td>
<td>1122527</td>
</tr>
<tr>
<td>First view</td>
<td>300405</td>
<td>365155</td>
<td>3000000</td>
</tr>
<tr>
<td>314567</td>
<td>315282</td>
<td>442137</td>
<td></td>
</tr>
<tr>
<td>670524</td>
<td>5559</td>
<td></td>
<td></td>
</tr>
<tr>
<td>30%</td>
<td>391</td>
<td>1324</td>
<td>13131</td>
</tr>
<tr>
<td>9892</td>
<td>123097</td>
<td>12393</td>
<td></td>
</tr>
<tr>
<td>300405</td>
<td>4011</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5559</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3000000</td>
<td>4011</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5559</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>30%</td>
<td>1171</td>
<td>2675</td>
<td>7099</td>
</tr>
<tr>
<td>14454</td>
<td>15365</td>
<td>75210</td>
<td></td>
</tr>
<tr>
<td>4877</td>
<td>5009</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6340</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7771</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Fig. 11. Progressive rendering of base LIDAR scan used in Patio scene

Fig. 12. Progressive and simultaneous rendering of four point clouds, with resolution increasing from top-left to bottom-right

Fig. 13. Rendering of LIDAR only (left) and LIDAR + still photo (right)

without the progressive refinement technique, the user would be waiting approximately this time to see anything. These values are similar to those we obtained with the similar technique presented in [30], despite this there are multiple point clouds (at least three) being downloaded simultaneously, and compare well with those state-of-the-art on the different but related problem of progressive mesh transmission [32]. The thumbnail images do not add overhead with respect to the 3D point cloud data, as their file sizes are comparatively small and they appear rapidly in the scene.

An interesting comparison of our progressive point cloud rendering method is with that provided by Potree [31]. While implementational details and timings of the Potree method are yet to be published, it clearly uses a similar level-of-detail approach to ours. However, beyond that basic similarity, the techniques appear different. Potree seems designed to minimize the data downloaded by increasing the level of detail of those areas which are currently within a certain distance of the camera. While our work does support this feature (see
Fig. 14. A (hidden) HTML5 video element pipes texture information, at 30 frames per second, positioned to the original camera location and orientation.

Fig. 15. Hybrid 2D-3D web-interface showing the timeline component and GUI overlaying the 3D context

[30]) we choose to disable it for this application, in the interest of downloading the entire dataset as quickly as possible - this also makes it unfeasible to compare download times, as Potree makes a point of not downloading the entire dataset if possible. We do note however, that in one of our trial datasets (the largest point cloud from the Patio set), the Potree rendering presents some artifacts between the cells of the hierarchical data structure, which are not present in our work (see Fig. 18).

IX. CONCLUSION AND FUTURE WORK

Typically, processing and visualisation of big multi-modal data is split between individual tools, with video and images processed in a 2D domain then visualised using a thumbnail browsing interface, and 3D data in dedicated 3D production and rendering software. In this paper, we have introduced a framework for unified 3D web-based visualisation of multi-modal digital media production datasets, which allows various input modalities to be registered into a unified 3D space, and visualised in hybrid-mode web application.

A multi-domain feature description extended from an existing feature descriptor and a hybrid RANSAC-based registration technique were proposed. The approach was tested on our multi-modal database acquired from various modalities including active and passive sensors as well as public single-modal dataset. The proposed method shows two times higher precision of feature matching and more stable registration performance than conventional 3D feature descriptors.

Visualisation of production data via the web is currently become increasingly relevant as modern workflows become based in the cloud. Our web-based visualisation takes advantage of the power of the web-context to integrate several viewing modalities into a single application, with the additional advantages of the web: machine independence, no specialised software requirements, viewing from anywhere in the world, etc. The results show that our progressive download method reduces the problems relating to remote viewing of big data. The principal contribution of this aspect of the work is that few other researchers have presented results on progressive visualisation of point cloud data via the web; and (to our knowledge) our work represents the first effort to do so as part of a wider hybrid visualisation of multi-modal data.

Future work on the multi-modal data registration aims to extend to a large-scale spatio-temporal scene data producing a coherent view of the world. It deals with synchronisation and registration of multi-modal data streams captured by very large and diverse collections of professional and consumer devices under uncontrolled and unpredictable environments. Another direction of extension will be registration of non-visual data such as audio and text (annotation and metadata). New feature description and matching method for cross-modalities should be developed. Although our current system works effectively on tablet devices, our future work on visualisation is now focused on integrating more elements of mixed reality into
the application. This possibility is opened due to the fact that mobile versions of many web browsers allow javascript access to the device accelerometer and camera, raising the prospect of remote users being able to visualise a current dataset in real-time (i.e. on the same day as the capture) and use of tablet devices as a virtual ‘window’ into the scene, moving it around in space to view the reconstructed scene.

REFERENCES


Hansung Kim received the MS and Ph.D degrees in electronic and electrical engineering from Yonsei University, Seoul, Korea, in 2001 and 2005, respectively. He was employed as a Researcher of Knowledge Science Lab (KSL) at Advanced Telecommunications Research Institute International (ATR), Japan, from 2005 to 2008. He is currently a Research Fellow (RA2) at the Centre for Vision, Speech, and Signal Processing (CVSSP) at the University of Surrey, Surrey, U.K. His research interests include 3-D computer vision, multi-modal data processing, audio-visual data processing and media production.

Alun Evans received a Ph.D. degree in medical physics from University College, London, U.K., in 2006. He is currently Lecturer in Graphics and Games at La Salle - Ramon Llull University, Barcelona, Spain. After completing postdoctoral work in computer graphics and character animation, he moved into the visual media production industry for several years, before rejoining academia in 2013 at Universitat Pompeu Fabra, where he was a member of the Interactive Technologies Group. He is currently with the GTM - Grup de Recerca en Tecnologies Mèdia, La Salle - Ramon Llull University, Barcelona, Spain, where he researches and teaches in the fields of 3D graphics and videogame development.

Josep Blat received the Ph.D. degree in mathematics from Heriot-Watt University, Edinburgh, U.K., in 1985. He is currently a Full Professor in the Department of Information and Communication Technologies at Universitat Pompeu Fabra, Barcelona, Spain. After his initial research in Applied Nonlinear Analysis, he moved to modeling and mathematical analysis of images (collective prize Philip Morris France 1991). His current research interests include advanced 3-D graphics (human modeling and animation, games), human-computer interaction (older people, ethnography, geolocation), and technology-enhanced learning (learning standards and tools, collaborative learning, new learning environments).

Adrian Hilton received the B.S.(Hons.) and D.Phil. degrees from University of Sussex, Sussex, U.K., in 1988 and 1992, respectively. He is currently a Professor of Computer Vision and Graphics and Director of the Centre for Vision, Speech, and Signal Processing at the University of Surrey, Surrey, U.K. His research interests include robust computer vision to model and understand real world scenes. Contributions include technologies for the first hand-held 3-D scanner, modeling of people from images and 3-D video for games, broadcast, and film production. He currently leads research investigating the use of computer vision for applications in entertainment content production, visual interaction, and clinical analysis.