2014 IEEE International Conference on Robotics & Automation (ICRA)
Hong Kong Convention and Exhibition Center
May 31 - June 7, 2014. Hong Kong, China

# Crowdsourced Saliency for Mining Robotically Gathered 3D Maps Using Multitouch Interaction on Smartphones and Tablets

Matthew Johnson-Roberson[1] and Mitch Bryson[2] and Bertrand Douillard[3]
and Oscar Pizarro[2] and Stefan B. Williams[2]

***Abstract*— This paper presents a system for crowdsourcing saliency interest points for robotically gathered 3D maps rendered on smartphones and tablets. An app was created that is capable of interactively rendering 3D reconstructions gathered with an Autonomous Underwater Vehicle. Through hundreds of thousands of logged user interactions with the models we attempt to data-mine salient interest points. To this end we propose two models for calculating saliency from human interaction with the data. The first uses the view frustum of the camera to track the amount of time points are on screen. The second treats the camera's path as a time series and uses a Hidden Markov model to learn the classification of salient and non-salient points. To provide a comparison to existing techniques, several traditional visual saliency approaches are applied to orthographic views of the models' photo-texturing. The results of all approaches are validated with human attention ground truth gathered using a remote gaze-tracking system that recorded the locations of the person's attention while exploring the models.**

## I. Introduction

We have developed a smartphone/tablet app for the viewing and manipulation of 3D models gathered with an Autonomous Underwater Vehicle (AUV). This app is freely available and has been download and used by a large number of users. The question this paper is attempting to answer is "Can we employ crowdsourcing to perform salient interest point detection from users not specifically tasked to find these points?" A diagram depicting the high-level system presented in this work is shown in Figure 1.

We are interested in saliency in the context of a long-term environment-monitoring program using AUVs. At the Australian Centre for Field Robotics there is an ongoing program to perform benthic monitoring with an AUV [1]. This program deploys an AUV in unstructured natural environments where it gathers data for human review. One of the major bottlenecks in this process is the vast amount of data gathered by the AUV. The AUV is capable of gathering orders of magnitude more data than previous techniques. Traditionally divers used hand-held cameras to gather visual data in underwater environments and issues of decompression, airtime, and safety severely limited the quantity of data

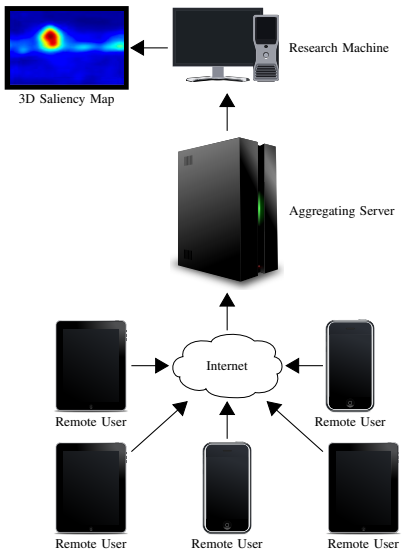



Fig. 1. Diagram of network architecture for crowdsourcing saliency.

that scientists could gather. With the AUV in its current configuration, monitoring images can be gathered at up to 4 Hz. A typical field campaign lasting two weeks can result in hundreds of thousands of images requiring review.

The challenge of how to deal with this massive image archive is being explored on several fronts. A large effort has gone into unsupervised clustering, human hand labeling, and supervised classification. This work presents an alternative for gathering large amounts of human review data quickly and inexpensively. The assertion we present in this paper is that human visual saliency can be modeled by proxy through the exploratory motions of a large number of users in a 3-D environment.

Capturing human curiosity and exploration for robotic platforms is non-trivial. The well established approach is to use visual saliency measures but it is not necessarily clear that they can predict what people find interesting in a 3D scene and how they will choose to interact with it. This paper presents two alternative measures of human interest both based on the motion of the viewpoint used by the operator and compares them to traditional saliency measures. Through the crowdsourcing of many remote smart phone/tablet users we gather data to perform the identification of visual saliency on 3-D photo mosaic maps. Human experiments with ground truth from eye tracking are used to validate our results.

Traditional crowdsourcing of vision tasks relies upon motivating users through community good will, financial incentives (Mechanical Turk) or competitive/entertainment

[1]Matthew Johnson-Roberson with the Department of Naval Architecture and Marine Engineering, University of Michigan, Ann Arbor, MI 48109 USA mattjr@umich.edu
[2]Australian Centre for Field Robotics University of Sydney, 2006, Australia {m.bryson,o.pizarro,stefanw}@acfr.usyd.edu.au
[3]Bertrand Douillard with the NASA Jet Propulsion Laboratory, California Institute of Technology, Pasadena, CA Bertrand.Douillard@jpl.nasa.gov

incentives [2] by turning a task into a game. Here we propose the use of a novel paradigm from big data analytics where the answers to questions can be inferred from the data of many users. The power of our data-mining approach to crowdsourcing is that data is collected from a much larger pool of users.

Using the smart phone platform gives us access to a much more general audience. To further the general appeal of the app we do not ask users to explicitly identify things they find interesting. Rather, we attempt to infer interest from patterns of interaction and in doing so free the user from an artificially constrained task. Without asking users to answer a specific question, their motivations for participating can be much more varied. This potentially gives access to a much larger 'crowd'.

We will be presenting two novel metrics to calculate saliency from human user interaction data. One employs the use of the camera's frustum to histogram observed points, while the other leverages a Hidden Markov model (HMM) to classify interaction data spatially into a saliency map. These techniques are compared to several state-of-the-art visual saliency techniques and validated using human gaze tracking data. The paper is laid out as follows. Section II discusses prior work. Section III presents the developed app as a platform for crowdsourcing. In Section IV the two interaction-based formulations for saliency are laid out. The human trials for validation are discussed in Section V. Results are presented in Section VI and finally Section VII concludes and presents future work.

## II. Prior Work

### A. Crowd Sourced Vision

Tools such as LabelMe, ImageNet, BUBL and other systems which leverage Amazon's Mechanical Turk have provided solutions to the problem of image-labeling using human computation [3]. Mechanical turk has become a particularly popular platform for crowd sourcing for vision. It offers flexibility and there has been research into assessing, processing, and rectifying image labelings from large groups of human sources [4]. All the aforementioned systems deal with image annotation and with various types of semantic information, including object identification, object classification, and object segmentation.

In the field of gaze tracking Rudoy et al. propose a relevant model of crowdsourcing gaze tracking. They project a pattern over video or image data. Then a 'crowd' of remote users enter the subsection of the pattern viewed providing a proxy for direct gaze tracking [5].

### B. Image Saliency

Research on the human perception system has shown that it is selective in its attention [6]. Human perception focuses on salient regions of an image and there has been a great deal of literature published proposing models for that process [7], [8]. Such work produces a saliency map for an image whereby the relative saliency of each pixel is expressed.

### C. Interaction-based saliency

Interaction-based saliency is the process of extracting saliency metrics from the way a user interacts with an image, video, or 3D model. Existing research on interaction-based saliency is primarily focused on 2D images and videos, not 3D models, however many relevant analogues exist. For 2D visual data Zoomable User Interfaces (ZUIs) or interfaces that allow users to zoom and pan around a large image or video have gained popularity in recent years. Utilizing these new interfaces several techniques have been developed to gather metadata for a piece of media being viewed in a ZUI. Carlier et al. propose a system where users watch a video and can retarget (zoom and center) it using the keyboard. Users input is then aggregated to produce a global retargeting for the video [9]. Similarly, Cricri et al. propose the use of implicit user region-of-interest data by detecting overlapping regions and concurrent events in multiple synced videos and attitude heading reference data streams [10].

For static images, one of the most relevant works to this study is that of Xu et al. who propose the use of 'Touch Saliency' by capturing the center and size of zooms to produce saliency maps which are compared to traditional image saliency measures using eye-tracking data [11]. In a non-touch based interface Baccot et al. proposed using a smart phone's picture browsing functionality to capture and store zoom and pan data. This data was then relayed to a central server to produce user interest maps [12].

## III. Mobile App

For this work we have created an app that allows users to explore and navigate a 3-D photo textured model of the seafloor. A screenshot of it running can be seen in Figure 2. Written in Objective-C and using OpenGL ES it runs on both phones and tablets. The app is downloadable for free [1]. Named *SeafloorExplore* the app was released into the Apple iTunes app store in 2012. The app itself uses virtual texturing [13] and a static Level-of-Detail (LOD) hierarchy to be able to display models of up to 1000 $km^2$ with textures up to $128k^2$ pixels. It is only more recent phone models that have the GPU capability to be able to display 3-D models of such size and scale. It is these advances that are driving the slow but gradual adaptation of 3-D maps on modern phones, replacing their 2-D analogs.

The app gathers and logs interaction data from users. This data is periodically relayed back to a central server. The data is aggregated and compressed to minimize network traffic.

### A. Model Generation

The models used in the app are created from data gathered in-situ beyond diver depths with an AUV. Once the AUV has completed an image gathering mission the vehicle is retrieved and the data downloaded. The AUV is equipped with a suite of navigation sensors: a 3-axis roll/tilt sensor, an acoustic Doppler Velocity Log (DVL), and a magnetic

[1] http://www-personal.acfr.usyd.edu.au/mattjr/app.html

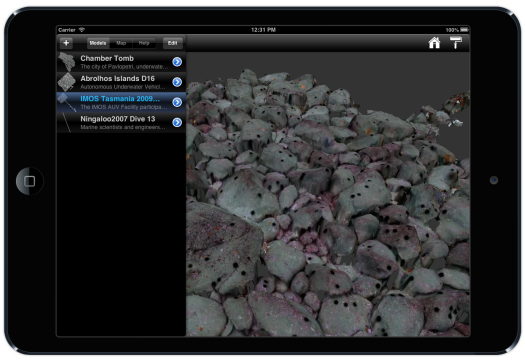



Fig. 2. Screenshot from the mobile app. The user can select from a variety of preinstalled models or download new models from a central server. There is the option to read a description of the context in which the data was gathered along with the scientific relevance of the area. The interaction takes place on the touch screen and the device's graphics processing unit (GPU) handles all the rendering of the geometry and textures. The app is capable of rendering models with textures up to $128k^2$ pixels by employing Virtual Texturing [13].

compass. These sensors along with the visual imagery are used to estimate poses for the vehicle.

The pose estimation process is performed in two steps. Firstly the data from cameras, DVL (with the AUV) and tilt sensor are fed to a Simultaneous Localization And Mapping (SLAM) filter. A sparse information-form pose-based SLAM algorithm [14] estimates the pose of the cameras for each pair of stereo images. Once this SLAM filter has completed we perform a second refinement step where the poses are optimized to further reduce the reprojection error. This second step takes the form of a sparse bundle-adjustment globally optimizing the poses given stereo matched robust features.

Once an optimized set of poses exists we employ the 3D reconstruction technique proposed by Johnson-Roberson, Pizarro, Williams, *et al.* [15]. We modify the previously proposed technique to operate in full 3D, preserving the visual quality of the model. Through the use of state-of-the-art model parameterization and texture atlasing the distortion of the final result can be minimized while the resolution of the original source imagery is maintained. A sample 3D model can be seen rendered on the screen of the iPad in Figure 2.

### B. Camera Motion

For multitouch interfaces gestures for 3D interaction are more variable across platforms and applications than for 2D interaction. We have selected a terrain centric interaction model which emphasizes the separation of degrees of freedom [16]. This model affords the user three gestures for three types of camera movement: pan, tilt/rotate, and zoom. Each type of motion is represented as a discrete value of the variable $m_t$. All operations occur around a point on the model which we will refer to as $P_0$. The camera is pointing at $P_0$ and is drawn back along a ray a distance $d$. The ray's angle with respect to the model both vertically and horizontally are expressed as $\lambda$ and $\phi$ respectively. This model is depicted in Figure 3.

Pan is performed using a single finger. When the user places their finger on the screen a ray is projected into the scene and intersected with the model. If the user's first touch point does not hit the model the ray is intersected with a plane spanning the $x$ and $y$ axes at the average $z$-depth of the model. This first intersection point forms the starting camera location for a model translation. As the user drags their finger we perform a camera transformation such that the original point remains under the user's finger, but the camera's center $P_0$ in Figure 3 is changed. This type of motion has an intuitive feel for an object like a terrain model.

Zoom is performed similarly to the two-dimensional case. A pinch gesture is used to change the distance $d$ of the camera along the ray between the center of projection and the model in Figure 3.

Tilt and rotate are both performed using a two-finger drag. We have chosen to separate each axis in x and y on the screen into tilt and rotate respectively. That is moving vertically on the screen modifies $\lambda$ and moving horizontally modifies $\phi$ as shown in Figure 3. The limits of tilt are stopped at 90 degrees to prevent going through the model.

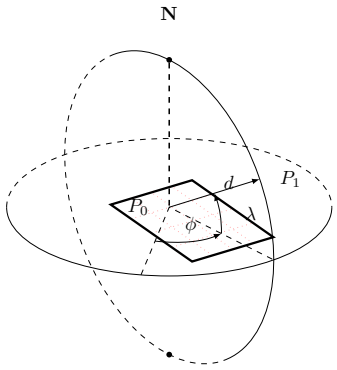



Fig. 3. Representation of spherical camera manipulator. The plane with the dotted grid represents the terrain. The camera model is oriented around a point $P_0$ on the terrain. The camera sits at $P_1$, located distance $d$ away from $P_0$. The angles of the ray with respect to the x and y axis between $P_0$ and $P_1$ is expressed by $\phi$ and $\lambda$ respectively.

One advantage of this camera model is that it is a compact representation of the camera position. The parameters to reconstruct the camera's movement:

$$\{m_t, P_0, d, \lambda, \phi, t\}$$

are stored continuously along with the time $t$. This data consists of seven 32-bit floats and one char $m_t$ (note $P_0$ is a vector in $\mathbf{R}^3$). It is important to keep a compact representation as this data is transmitted (often over low-bandwidth mobile internet) back to a central server. Once aggregated on the central server it can be downloaded and used as the source data for the proposed saliency detection algorithm.

### C. Crowdsourcing Participants

The app *SeafloorExplore* has been available in the Apple iTunes app store since June 2012. Without any additional

advertising beyond what occurs automatically in the app store it has gathered over 4,500 unique users from all over the world. Each month between 500 and 1000 sessions occur each gathering interaction data. Eight thousand total sessions have been logged. Furthermore there have been over 350,000 total camera movement events recorded and transmitted to the central server. On average 30.47 events are logged per session. This has, without direct request for labeling, generated a massive data set for the saliency analysis.

## IV. Interaction-based Saliency Methods

Building upon the work in interaction-based saliency discussed in Section II-C this section will propose two novel frameworks that attempt to capture the notion of saliency using 3D camera motions (as described in Section III-B) crowdsourced from the developed app. While traditional visual image saliency experiments rely on static images, this paper presents two key differences to that work. First, 3-D photo textured models are used which means that not only does intensity and color play a role in saliency, but depth and relief contribute to what is or is not interesting. Second, the maps employed in this work can be thought of as multiscale. The resolution and extent of the maps is larger than could be captured by the eye or the screen in any single view. With such maps there is a relationship between a low-resolution overview and a high-resolution zoom that the user explores by navigating the model. Very simply we make the assumption that the user will attempt to see a higher resolution version of something they are interested in. To this end the multitouch interface (described in Section III-B) along with the virtual texturing renderer (see Section III) allow for the exploration of the model at arbitrary resolutions.

One key assumption of this work is that users are looking at what they find interesting. In a static single image set up, when the users gaze is tracked, this assumption holds very well. We assert that while this assumption may not hold for any single user selected at random from a pool of remote users it does hold across the group. We assume on average most people are looking at what they find interesting. Usage patterns that violate this assumption do exist. A user may be attempting to learn the interface and simply performing actions to understand the mapping between touch and motion. A user may be moving randomly not understanding or interested in what they are looking at. And finally a user may have a gaze pattern that is very uncorrelated with motion. In the following section we will attempt to show these usage patterns are not the dominant trend in the data and that with a sufficiently large amount of users such patterns do not affect the outcome of the analysis.

### A. Frustum Based Saliency

The first proposed technique which we will refer to as the *frustum* method uses the camera's view of the scene to model human interest. This idea is based on the simple principal that users will move the camera to cover areas of the scene they are interested in looking at. The more times a point appears in view the more 'salient' it is said to be. To begin the algorithm we initialize every point in the scene to have a counter of zero. The camera parameters recorded from the app are used to move around a virtual camera with the same projective parameters as the user saw in the viewer. This camera is swept over the model and each time a point is within the camera's frustum we increment its counter. The camera's motion is discretized by a constant time unit so the longer the camera views a set of points the greater the increment of the counter. After the camera paths from all users have been replayed with the virtual camera we spatially histogram the points to produce a saliency map. This trivial algorithm produced quite compelling results, however a more complex model of the camera's motion is presented in the following section.

### B. Hidden Markov Model Saliency

Here we present a second technique to capture the notion of saliency from interaction. We propose the use of a Hidden Markov model to capture the motion of the camera through time. Hidden Markov models (HMMs) are probabilistic state machines that have been used extensively for the classification of time series data. Most notably HMMs have been used for speech recognition and bioinformatics to great success [17]. We apply them here to give us a formulation that can learn the relationship between motions which indicate interest: such as spending a long time rotating around a point, zooming in and out of a point, or tilting about a point. To capture these more complex chaining of motions the camera's path can be thought of as a time series. Through this time series the user is either interested in a single point or moving around looking for something new to explore. The amount of time spent on a point and the motions that occur near it provide a strong indicator of the user's intentions with respect to that point.

### C. Structure of a HMM

In a HMM the Markov property is assumed for the modeled system. The model contains a sequence of observable data and a set of unobserved states. In this formulation the unobserved or hidden states correspond to the classification of saliency, as either salient or non-salient. The input, observable data, is in this case the camera motions gathered from a user. A HMM is a probabilistic model and two sets of probabilities determine the classification of the input data. The first are the observational probabilities. These probabilities describe the likelihood that an observation corresponds to a certain state. Second are the transitional probabilities. These probabilities describe the likelihood of a transition between two states.

For the classification of saliency a two state HMM is proposed. A diagram of the proposed two state HMM is shown in Figure 4. A two-state HMM consists of two random variables $O$ and $Y$ which are both sequences. The variable $Y$ consists of states $\{y_0, y_1, ..., y_n\}$. $Y$ changes sequentially with the first order Markov property. This means

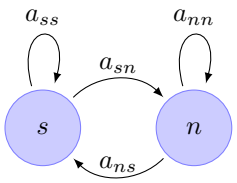



Fig. 4. An HMM with 2 states salient ($s$) and non-salient ($n$). In this particular HMM, states can only reach themselves or the adjacent state.

the probability of a state change in $Y$ only depends on its current state.

Formally:

$$\begin{aligned} P(Y(t+1) = y_i | Y(0)...Y(t)) \\ &= P(Y(t+1) = y_i | Y(t)) \\ &= P(Y(2) = y_i | Y(1)) \end{aligned}$$

These states can take two values $s$ (salient) and $n$ (non-salient).

Formally the hidden state $Y$ is:

$$Y = \begin{cases} s & \text{salient} \\ n & \text{non-salient} \end{cases}$$

The observable data is $O$ with states $\{o_0, o_1, ..., o_m\}$. Here it is the camera motion feature data as described in Section IV-D.

Using Bayes' rule:

$$P(Y|O) = \frac{P(O|Y)\,P(Y)}{P(O)}$$

In the maximization of $P(Y|O)$ the constant $P(O)$ can be eliminated and the optimal $Y$ (denoted $\hat{Y}$) can be computed using:

$$\hat{Y} = \underset{Y}{\operatorname{argmax}} \underbrace{P(O|Y)}_{\text{observation}} \cdot \underbrace{P(Y)}_{\text{transition}}$$

The transition probabilities $P(Y)$ are parameterized by a $2 \times 2$ matrix, A:

$$A = \begin{pmatrix} a_{ss} & a_{sn} \\ a_{ns} & a_{nn} \end{pmatrix}$$

as the following notation generalizes to an $n$ state HMM we will refer to the $s$-state as state 1 and the $n$-state as state 2 therefore $a_{ij}$ in $A$ refers to the probability of transitioning between state $i$ and $j$ where these indices correspond to the rows and columns of $A$.

The observation probabilities $P(O|Y)$ are defined by a $m \times n$ dimensional matrix $B$ containing the observation probabilities such that

$$B(i,j) = P(O = o_i | Y = y_j).$$

Every state $j$ has an associated probability distribution $b_j(o_t)$ which is the probability of generating observation $o_t$ at time $t$.

The representation for the probability density function (pdf) that is used is a mixture of higher-dimensional Gaussian distributions:

$$b_j(o_t) = \sum_{m=1}^{M} c_{jm} \times \mathcal{N}(o_t, \vec{\mu}_{jm}, \Sigma_{jm})$$

where $M$ is the number of observations in each state, $c_{jm}$ is the mixture coefficient for the $m$-th mixture in state $j$, $\vec{\mu}_{jm}$ is the mean vector for the $m$-th mixture component in state $j$, $\Sigma_{jm}$ is the covariance matrix for the $m$-th mixture component in state $j$, and $\mathcal{N}(o_t, \vec{\mu}_{jm}, \Sigma_{jm})$ is a multi-dimensional Gaussian (Normal) distribution.

The classification of $\hat{Y}$ is performed in standard fashion using the Viterbi algorithm [17].

### D. Camera Feature Data

To parametrize the camera data a model that represents the camera's velocity is used. The intuition is that the user will slow the camera down with respect to a point on the surface of the model that they are interested in. The camera model (described in Section III-B) has a parameter $P_0$ which is the point which the center projective ray of the camera passes through. Differencing this parameter over time the velocity vector $v$ of the camera at time $t$ in $m/s$ can be calculated and this is the observation $o_t$ used in the HMM.

### E. Training

In the HMM formulation described above the parameters require training. The parameters to be trained are $\lambda$:

$$\lambda = (A, B)$$

One option for training is to use labeled data to find a set of parameters $\lambda$ that maximize the correct classification. This option was explored as there is correlated eye tracking and motion data from the human validation experiments described in Section V. This data can be treated as a labeling by using fixation points as positive examples of saliency. However, techniques that do not require labeled data are regularly used to train HMMs. This option was selected as performance differences were negligible. Additionally, such techniques remove the need to conduct eye tracking experiments to implement the algorithm.

To train without labeled data the Baum-Welch algorithm maximizes $P(Y|\lambda)$ with respect to $\lambda$ [18]. The algorithm performs a local search and is therefore sensitive to initial parameters. The result is a set of parameters optimized to explain the input data. A full discussion of this process can be found in Rabiner [17].

## V. Experiments

To validate that the proposed technique is an effective proxy for human interest we set up a traditional visual saliency eye tracking experiment. Eye tracking has long been used in psychology, human computer interaction, and vision research to experimentally measure human attention [19].

### A. Experimental Setup

The eye tracking experiment had the following form. A near-infrared remote eye tracking system that offered sub degree resolution on the user's gaze location was placed in a fixed position. The gaze tracker logs continually and transmits the current gaze location over TCP/IP. The user sits at a desk with an iPad permanently mounted above the gaze tracking equipment. The system is calibrated to return gaze locations on the iPad's screen. The iPad reports all of its interaction and model rendering data directly to the same logging computer over User Datagram Protocol (UDP). The user is given a test model to help them familiarize themselves with the interface. The user has two minutes to learn to navigate the interface using the test model (a randomly generated fractal terrain). After the test period the user is presented with three models and has three minutes to explore each in sequence. The time period of three minutes was selected as this is the average approximate time users interact with any model as computed from statistics gathered from the app. Twelve users participated producing a total of 313,161 total fixation points across all the models tested.

### B. Saliency Comparison Techniques

We compare our saliency maps generated from interaction to the following published saliency algorithms: the original Itti-Koch saliency model (denoted Itti) [20], Graph-Based visual saliency (denoted GBVS) [21], Rudinac's salient object detection algorithm (denoted Rudinac) [22], and Hou's spectral saliency algorithm (denoted Hou) [23]. These methods provide another means of generating a saliency map using an orthographic image of the 3D model's visual texture as their input. As such methods are commonly the way saliency is calculated we feel they provide an important comparison to this new technique.

## VI. Results

The results are generated across data gathered on missions from three separate field deployments: The first dataset (denoted Geebanks in figures) was gathered at a site on the Western Australian coastline, the Geebank area of the Abrolhos Islands. The second dataset (denoted Ningaloo in figures) was gathered at the Ningaloo Marine Park located in the Ningaloo Reef on Australia's northwest coast. Finally the third dataset (denoted St. Helens in figures) was gathered in Tasmania, Australia off the coast of St. Helens. Here sea urchins have invaded the local habitat and formed barrens disrupting the indigenous ecosystem.

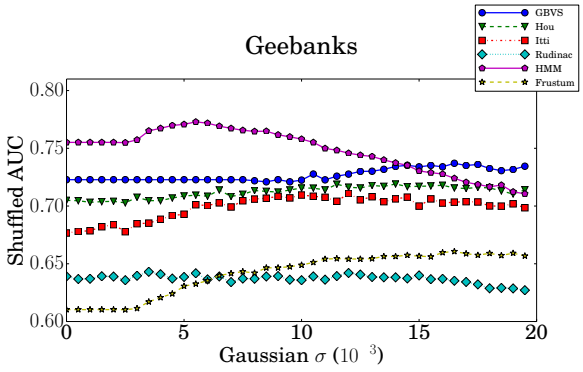
We present a quantitative comparison with the same bottom-up visual saliency measures again using human gaze tracking data as ground truth. Results are shown using the shuffled Area Under Curve (AUC). The shuffled AUC is a slight modification of the traditional AUC for Receiver Operating Characteristics (ROC) curves. The problem of visual saliency can be thought of as a binary classification task where the decision boundary is between salient and non-salient. In this formulation human fixations are considered the positive set while random points from the image are sampled to form the negative set. Perfect classification is a AUC of $1.0$ while random chance is $0.5$. Because our image mosaics are non-rectangular and only pixel data from the actual mosaic and not the black background is used. This prevents the border areas from biasing the results. As discussed by Zhang et al. border effects and center bias can significantly affect results [24]. The shuffled AUC selects the negative set from the union of all fixation points across the data set with the exception of the positive set. The shuffled AUC is gaining popularity as a metric to assess saliency measures [25].

As noted by Hou et al. one parameter that can have a significant effect on model accuracy is the smoothing or blurring of the saliency map [26]. Traditionally this smoothness is achieved by convolving the map with a Gaussian kernel. To assess each technique's sensitivity to this parameter the $\sigma$ or width of the Gaussian kernel is varied (from $5 \times 10^{-4}$ to $2 \times 10^{-2}$ times the image size in steps of $5 \times 10^{-3}$) [25]. The result of this comparison run across the three sets of field data can be seen in Figure 5. In the Geebanks results (Figure 5(a)) most of the visual saliency techniques perform fairly similarly when compared using this metric. However, the proposed HMM technique consistently exceeds all other techniques for low $\sigma$ and has the highest peak shuffled ROC at 0.772 and when tested under a varying decision boundary and randomized sample locations, performs quite well. The frustum based technique sees a performance bump from increasing $\sigma$ suggesting again that its localization of salient features is quite poor as blurring improves the results.

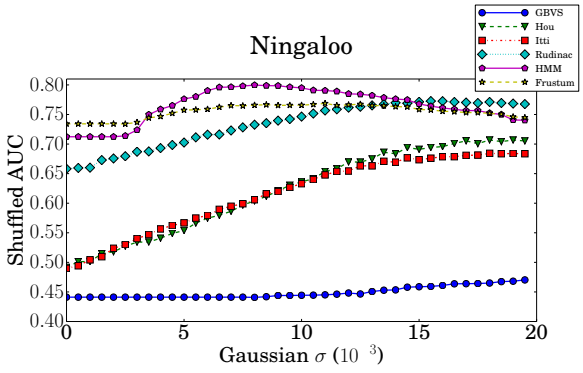
The result of the Ningaloo data set appears in Figure 5(b) here we see the strength of the frustum based algorithm on a flat thin dataset. The frustum technique is not hampered by 'incidental contact' in this dataset. When looking obliquely at the model very few points in front or behind the object of interest are in view. This leads to an up-weighting of saliency for primarily the object not the background. Performance for the frustum technique starts and stays high (peaking at an AUC of 0.767) across all tested $\sigma$ values. Again here we see the strength of the proposed HMM technique with the highest peak performance (almost 0.8 AUC). Additionally this score is for well localized maps with smaller $\sigma$. Several of the visual techniques perform well, notably the rudinac object detector and the Itti-Koch and Hou models for large $\sigma$.

Finally the St. Helens dataset (Figure 5(c)) shows the HMM technique again leading the pack for small $\sigma$ and again achieving the highest peak performance (0.769 AUC). However, its performance becomes comparable to the visual saliency techniques for larger $\sigma$. The mix of high relief and flat and thin structure is challenging and lead to the poor performance for the Itti-Koch model.

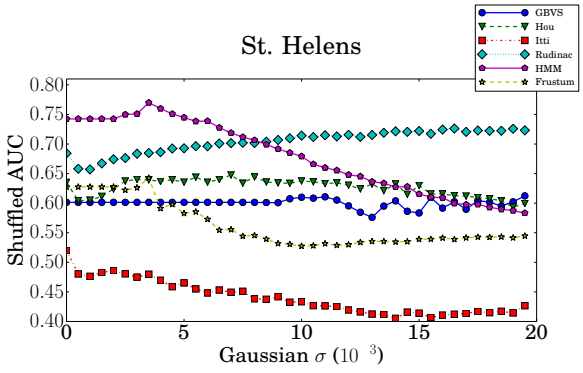
A single run $\sigma = (0.006 \times I_{size})$ is shown in Figure 6 here the error bars express the standard deviation across the multiple random point shuffles runs. Note the superior performance of the proposed HMM technique across all datasets.




(a)




(b)




(c)

Fig. 5. The Shuffled Area Under the Curve (AUC) scores for the AUC of the Receiver operating characteristic curves (ROC) for saliency maps computed with the proposed technique and four traditional visual saliency techniques. An important parameter for generation of these saliency map in all the techniques is $\sigma$ which is the size of the Gaussian smoothing kernel. These plots display the effect of varying $\sigma$ where $\sigma$ from $5 \times 10^{-4}$ to $2 \times 10^{-2}$ times image size in steps of $5 \times 10^{-3}$ over all techniques. Each graph is a separate data set gathered with an Autonomous Underwater Vehicle. (a) is from the Geebanks area near the Abrolhos Islands in Western Australia. (b) is the from the Ningaloo Reef in Western Australia. And (c) is from an sea-urchin infested boulder field in Tasmania, Australia. The performance of the two proposed techniques *Frustum* and *HMM* (using only interaction data) is comparable to all the tested visual saliency metrics from the literature.

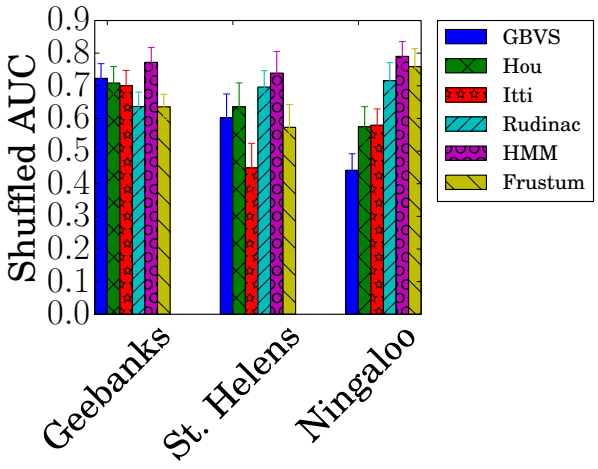



Fig. 6. Shuffled Area Under the Curve (AUC) scores for the AUC of the Receiver operating characteristic curves (ROC) for saliency maps computed with the proposed technique and four traditional visual saliency techniques. Here human gaze tracking is used as ground truth. Three datasets (3D maps gathered with an AUV) are presented. These map vary in structure and visual content but the proposed techniques perform well across each. An important parameter for generation of these saliency map in all the techniques is $\sigma$ which is the size of the Gaussian smoothing kernel. Here $\sigma = (0.006 \times I_{size})$ a function of image size.

## VII. Conclusions and Future Work

In this paper we have presented a novel technique for extracting saliency from crowdsourced interaction data. We have developed two algorithms to extract interest metrics from camera motions. To our knowledge this system is the first of its kind in using a distributed smart phone app to gather saliency data for 3D maps. The proposed system provides an alternative to traditional visual saliency by harnessing the modality of touch to provide a proxy for interest. Our results show that comparable performance to visual saliency is achievable solely through interaction given a large enough user base. Through a gaze tracking experiment we have validated the techniques showing the proposed metrics are consistent with human attention.

The level of agreement between the human experiment and the automated techniques is good, however looking at the human subject data suggests the variability of 'interest' as a definition for any saliency metric. The process that governs what we as humans find interesting in a large 3D map is more complex than the traditional model of visual attention for a static image. Additionally because of the amount of time spent exploring the model this process is likely to evolve. People will become more or less interested in certain types of organisms or terrain structures over time. This means the path of a human through the model, especially a large model is rarely going to be similar for small trial sizes.

This points to the challenge of applying any saliency technique on a large, visually and structurally rich model. Casual users will not explore a model with the rigor or depth of a scientist reviewing images for scoring. As such the proposed technique can not replace the ongoing efforts to automate the segmentation and classification of such science

data. However the technique offers a promising approach for discovery and data-mining in a vast archive of images (currently we hold an archive of over a million AUV images).

Ultimately the ability to understand what users find interesting in 3D models is appealing for applications beyond the one presented here. Mapping companies, advertisers, and geo-statisticians are all interested in what people are looking at on a map. As 3D maps overtake their 2D counterpoints on smart phones, interest metadata could help businesses know what streets are explored commonly, or what terrain features people are drawn to in a region.

As we move forward a greater leveraging of the existing work in how to intelligently rectify the labels of multiple operators when crowdsourcing would strengthen the approach presented. The use of more complex inter-operator relationships in terms of cross checking and aggregation of interest could further refine the saliency maps produced. A more complex HMM model could capture more states such as exploratory motion vs. directed search in addition to the salient/non-salient distinction. Also exploring the use of the depth information as a saliency channel could improve the performance of the visual only techniques.

## REFERENCES


[1] S. Williams, O. Pizarro, M. V. Jakuba, C. R. Johnson, N. S. Barrett, R. C. Babcock, G. A. Kendrick, P. D. Steinberg, A. J. Heyward, P. Doherty, I. Mahon, M. Johnson-Roberson, D. Steinberg, and A. Friedman, "Monitoring of benthic reference sites using an autonomous underwater vehicle," *IEEE Robotics and Automation Magazine*, vol. 19, no. 1, pp. 73–84, 2012.

[2] L. von Ahn and L. Dabbish, "Labeling images with a computer game," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ser. CHI '04, Vienna, Austria: ACM, 2004, pp. 319–326.

[3] N. Kumar, A. Berg, P. Belhumeur, and S. Nayar, "Attribute and simile classifiers for face verification," in *Computer Vision, 2009 IEEE 12th International Conference on*, 2009, pp. 365 –372.

[4] V. C. Raykar, S. Yu, L. H. Zhao, G. H. Valadez, C. Florin, L. Bogoni, and L. Moy, "Learning from crowds," *J. Mach. Learn. Res.*, vol. 99, pp. 1297–1322, 2010.

[5] D. Rudoy, D. B. Goldman, E. Shechtman, and L. Zelnik-Manor, "Crowdsourcing gaze data collection," *CoRR*, vol. abs/1204.3367, 2012.

[6] D. Walther and C. Koch, "Modeling attention to salient proto-objects," *Neural Networks*, vol. 19, no. 9, pp. 1395 –1407, 2006.

[7] G. Deco and B. Schürmann, "A hierarchical neural system with attentional top-down enhancement of the spatial resolution for object recognition," *Vision Research*, vol. 40, no. 20, pp. 2845–2859, 2000.

[8] L. Itti, C. Koch, and E. Niebur, "A model of saliency-based visual attention for rapid scene analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 11, pp. 1254–1259, 1998.

[9] A. Carlier, V. Charvillat, W. T. Ooi, R. Grigoras, and G. Morin, "Crowdsourced automatic zoom and scroll for video retargeting," in *Proceedings of the international conference on Multimedia*, ser. MM '10, Firenze, Italy: ACM, 2010, pp. 201–210.

[10] F. Cricri, K. Dabov, M. J. Roininen, S. Mate, I. D. D. Curcio, and M. Gabbouj, "Multimodal semantics extraction from user-generated videos," *Adv. MultiMedia*, vol. 2012, 1:1–1:1, Jan. 2012.

[11] M. Xu, B. Ni, J. Dong, Z. Huang, M. Wang, and S. Yan, "Touch saliency," in *Proceedings of the 20th ACM international conference on Multimedia*, ser. MM '12, Nara, Japan: ACM, 2012, pp. 1041–1044.

[12] B. Baccot, V. Charvillat, R. Grigoras, and C. Plesca, "Visual attention metadata from pictures browsing," in *Image Analysis for Multimedia Interactive Services, 2008. WIAMIS '08. Ninth International Workshop on*, 2008, pp. 122 –125.

[13] M. Mittring and Crytek GmbH, "Advanced virtual texture topics," in *ACM SIGGRAPH 2008 Games*, ser. SIGGRAPH '08, Los Angeles, California: ACM, 2008, pp. 23–51.

[14] I. Mahon, S. Williams, O. Pizarro, and M. Johnson-Roberson, "Efficient view-based slam using visual loop closures," *Robotics, IEEE Transactions on*, vol. 24, no. 5, pp. 1002–1014, 2008.

[15] M. Johnson-Roberson, O. Pizarro, S. B. Williams, and I. Mahon, "Generation and Visualization of Large-scale Three-dimensional Reconstructions from Underwater Robotic Surveys," *Journal of Field Robotics*, vol. 27, no. 1, pp. 21–51, 2010.

[16] A. Martinet, G. Casiez, and L. Grisoni, "The effect of dof separation in 3d manipulation tasks with multi-touch displays," in *Proceedings of the 17th ACM Symposium on Virtual Reality Software and Technology*, ser. VRST '10, Hong Kong: ACM, 2010, pp. 111–118.

[17] L. R. Rabiner, "A tutorial on hidden markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–286, 1989.

[18] L. R. Welch, "Hidden markov models and the baum-welch algorithm," *IEEE Information Theory Society Newsletter*, vol. 53, no. 4, 2003.

[19] C. M. Privitera and L. W. Stark, "Algorithms for defining visual regions-of-interest: comparison with eye fixations," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 9, pp. 970–982, 2000.

[20] L. Itti, C. Koch, and E. Niebur, "A model of saliency-based visual attention for rapid scene analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 11, pp. 1254–1259, Nov. 1998.

[21] J. Harel, C. Koch, and P. Perona, "Graph-based visual saliency," in *Advances in Neural Information Processing Systems 19*, MIT Press, 2007, pp. 545–552.

[22] M. Rudinac and P. Jonker, "Saliency detection and object localization in indoor environments," in *Pattern Recognition (ICPR), 2010 20th International Conference on*, 2010, pp. 404 –407.

[23] X. Hou and L. Zhang, "Saliency detection: a spectral residual approach," in *Computer Vision and Pattern Recognition, 2007. CVPR '07. IEEE Conference on*, 2007, pp. 1 –8.

[24] L. Zhang, M. H. Tong, T. K. Marks, H. Shan, and G. W. Cottrell, "Sun: a bayesian framework for saliency using natural statistics," *Journal of Vision*, vol. 8(7), no. 32, pp. 1–20, 2008.

[25] A. Borji and L. Itti, "Exploiting local and global patch rarities for saliency detection," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Providence, Rhode Island*, 2012, pp. 1–8.

[26] X. Hou, J. Harel, and C. Koch, "Image signature: highlighting sparse salient regions," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 34, no. 1, pp. 194 –201, 2012.