2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
Congress Center Hamburg
Sept 28 - Oct 2, 2015. Hamburg, Germany

Entropy Based Keyframe Selection for Multi-Camera Visual SLAM

Arun Das* and Steven L. Waslander†

Abstract— **Although many state-of-the-art visual SLAM algorithms use keyframes to help alleviate the computational requirements of performing online bundle adjustment, little consideration is taken for specific keyframe selection. In this work, we propose two entropy based methods which aim to insert keyframes that will directly improve the system's ability to localize. The first approach inserts keyframes based on the cumulative point entropy reduction in the existing map, while the second approach uses the predicted point flow discrepancy to select keyframes which best initializes new features for the camera to track against in the future. We implement the proposed methods within the Multi-Camera Parallel Mapping and Tracking framework, and demonstrate the effectiveness of our methods using ground truth data collected using an indoor positioning system.**

I. Introduction

Visual SLAM algorithms have been used successfully in a wide array of applications, ranging from inspection and surveillance [1], [2], to outdoor and Martian exploration [3], [4]. Accurate positioning is generally made possible by taking multiple measurements of visual landmarks throughout the trajectory of the vehicle, refining the landmark locations using a batch optimization procedure, and tracking against the landmark locations. Although effective for localization, processing image features and performing batch optimization over a large number of parameters requires substantial computation.

To that end, visual mapping techniques often use keyframes in order to reduce the problem size of the batch optimization. Existing approaches generally insert keyframes based on point triangulation baseline [5] or the overlap in the number of tracked points [6]. These heuristics attempt to insert keyframes in order to maintain the map integrity, yet do not directly attempt to minimize the uncertainty in the map. The approach detailed in [7] presents a method for variable baseline stereo, where the triangulation baseline is selected to minimize the modeled stereo depth error. Although the approach could be adopted to operate with keyframes, the formulation does not allow one to consider measurements from more than two view-points, camera motion, or initialization of new points into the map. The work most related to ours generates image features off-line, creates a buffer of the image frames, and selects keyframes in order to reduce content redundancy [8]. In contrast, our approach is a real-time, online system, and attempts to reduce map point *entropy* using an information theoretic (IT) formulation. Existing IT based methods [9], [10], focus on using mutual information between points to improve feature matching and map partitioning, whereas we directly target the keyframe selection problem.

In this work, we propose two separate entropy based keyframe selection methods which aim to accomplish two different tasks. First, we formulate an approach which seeks to select keyframes based on the *expected cumulative point entropy reduction (CPER)* in the currently existing map. Second, we propose an approach which seeks to select keyframes which are expected to initialize the most favorable *new* feature points, given predicted camera motion over a finite time horizon. The new points are evaluated using the *point pixel flow discrepancy (PPFD)* between the currently existing map points and the predicted location of new points triangulated from multiple views. The CPER approach is well suited for inspection, surveillance, or other tasks which require precise localization in a fixed region, while, PPFD is better suited for exploration tasks.

We demonstrate the benefit of employing entropy based keyframe selection on the multi-camera parallel tracking and mapping (MCPTAM) framework. MCPTAM accommodates multiple, non-overlapping wide FOV cameras by employing the Taylor camera model [11]. The added flexibility of the Taylor model, however, makes keyframe selection much more challenging as features observed in different parts of the image contribute very differently to localization accuracy. The typical keyframe selection methods are, therefore, not well suited to the MCPTAM framework, making it an excellent test case for the CPER and PPFD approaches. Although we demonstrate our methods using the MCPTAM framework, a similar formulation can be developed for any visual slam algorithm which uses keyframes [12], [6].

II. Entropy Computations

The Shannon entropy is a measure of the unpredictability or uncertainty of information content [13]. In the case where the probability density function of a continuous random variable, Y, is modeled as a Gaussian distribution, the Shannon entropy is $h_e(Y) = \ln(\sigma\sqrt{2\pi e})$, where σ is the variance of the distribution, and $h_e(Y)$ is used to denote the natural logarithm, in units of nats. Similarly, the entropy of a multivariate Gaussian distribution can be computed as

$$h_e(Y) = \frac{1}{2}\ln((2\pi e)^n |\Sigma|), \tag{1}$$

where Σ is the covariance matrix of the multivariate Gaussian distribution, and $|\cdot|$ denotes the determinant operator. Note that unlike the entropy of discrete random variables, it is possible for the entropy of continuous random variables to be less than zero.

* Ph.D. Student, Mechanical and Mechatronics Engineering, University of Waterloo; adas@uwaterloo.ca.

† Assistant Professor, Mechanical and Mechatronics Engineering, University of Waterloo; stevenw@uwaterloo.ca

Relative Entropy: The relative entropy (also called the *Kullback-Leibler divergence*) provides a measure of the difference between two probability distributions. If we denote two discrete probability distributions, X and Y, then the relative entropy between the two is

$$\mathcal{D}(X\|Y) = \sum_i P(x_i) \ln \frac{P(x_i)}{P(y_i)}. \quad (2)$$

The relative entropy measures the information lost when Y is used to approximate X, and is zero if and only if $X = Y$.

III. Multi-Camera Parallel Tracking and Mapping

MCPTAM [14] is a real-time, feature-based, visual SLAM algorithm which extends Klein and Murray's Parallel Tracking and Mapping (PTAM) [5] in numerous ways. It accommodates multiple, non-overlapping field-of-view (FOV), heterogeneous cameras in any fixed configuration. It also extends the PTAM's pinhole camera model to work with fish-eye and omnidirectional lenses through the use of the Taylor camera model [11], which prevents feature starvation due to occlusions and textureless frames in any single camera. Finally, MCPTAM introduces both an improved update process based on box-plus manifolds, and a novel feature parameterization using spherical co-ordinates anchored in a base-frame [15] for bundle adjustment.

A brief formulation of MCPTAM is given as follows: Denote a point in the global frame, $p \in \mathbb{R}^3$ as $p = [p_x \ p_y \ p_z]^T$ where p_x , p_y , p_z represent the x, y, and z components of the point, respectively. Let the map P be a set of points, defined as $P = \{p_1, p_2, \ldots, p_n\}$. Denote the reprojection function as $\Pi : \mathbb{R}^3 \mapsto \mathbb{P}^2$, which maps a point in the global 3D frame, $p_i \in \mathbb{R}^3$, to a pixel location on the image plane, $u_i \in \mathbb{P}^2$. In order to accommodate the large radial distortion caused by fisheye lenses, the Taylor model uses a spherical mapping, where the elevation and azimuth angles to a 3D point, $s = [\theta, \phi]$, are modeled as half lines which pass through the sphere's center, and finally mapped to the image plane through a polynomial mapping function and affine transformation [11].

In order to track the camera cluster pose parameters, $\omega \in \mathbb{SE}^3$, the map points, P, are reprojected into the image frames of the cameras, matched through patch warping to the current image features, and a nonlinear least squares solver updates the pose through reprojection error minimization. The map is constructed through bundle adjustment using *multi-keyframes*, which are a snapshot of all the images and point measurements taken from a pose along the camera cluster's trajectory. We shall define a multi-keyframe, M, as collection of keyframes, $M = \{K_1, K_2, \ldots, K_m\}$, corresponding to the m individual cameras which are part of the multi-camera cluster.

IV. Proposed Approaches for Keyframe Selection

To determine the best multi-keyframe to include from a buffer of existing options, we propose two main selection metrics: the cumulative point entropy reduction (CPER), which maximizes the expected entropy reduction in the existing map points, and the point pixel flow discrepancy (PPFD), which assesses the expected future features that can be added to the map.

A. Selection Based on Cumulative Point Entropy Reduction

In order to determine when a keyframe should be inserted into the map, we inspect the uncertainty of the current camera cluster provided by the tracking process. The covariance of the tracking pose parameters is given by $\Sigma^c = (G^T W G)^{-1}$, where $G = \frac{\partial \Pi}{\partial \omega}$ is the Jacobian of the map reprojection error with respect to the cluster state, and W is the matrix of weights associated with the measurements. To assess the current tracking performance, we extract the diagonal elements of covariance matrix Σ^c, which reflect the individual uncertainty of each of the tracker state parameters, σ_x, σ_y, σ_z, σ_{r_x}, σ_{r_y}, and σ_{r_z}, where x, y, and z are the translational parameters according to the x, y, and z directions, and r_x, r_y, and where r_z denote the rotations about the x, y, and z axis, respectively. Finally, keyframe addition is triggered when the entropy of any element of the positional entropy is above a user-defined threshold, ϵ, or $\max(h_e(\sigma_x), h_e(\sigma_y), h_e(\sigma_z), h_e(\sigma_{r_x}), h_e(\sigma_{r_y}), h_e(\sigma_{r_z})) > \epsilon$, where $h_e(\cdot)$ is computed using Equation 1. Multi-keyframe candidates are maintained in a buffer and scored based on the expected reduction in point depth entropy, if added to the map through a bundle adjustment process.

Suppose the tracking thread is currently operating at time t, and the last multi-keyframe insertion occurred at time k. Denote the set of multi-keyframe candidates, which are buffered between times t and k, as $\Phi = \{M_t, M_{t-1}, M_{t-2} \ldots, M_{t-k}\}$. Since each of the multi-keyframe candidates are saved from the tracking thread, an estimate of the global pose of each candidate is available from the tracking solution. Denote the set of map points from P, visible in $K_l \in M_i$, as $\tilde{P}_{K_{il}} \subset P$.

In the bundle adjustment process, the point parameters are modeled as a Gaussian distribution with an associated mean and covariance, denoted as $\hat{p}_j$ and $\Sigma_j \in \mathbb{R}^{3\times3}$, respectively. Suppose point $p_j \in \tilde{P}_{K_{il}}$ is measured in keyframe $K_l \in M_i$. Our method seeks to determine the updated covariance of point p, if triangulated using an additional measurement from keyframe K_l. This is accomplished using a covariance update step, similar to the Extended Kalman Filter.

Denote the Jacobian of the reprojection function, with respect to the point parameters p, evaluated at point $\hat{p}_j$, as

$$J_j = \frac{\partial \Pi}{\partial p}|_{\hat{p}_j}. \quad (3)$$

Using the Jacobian, J_j, and the prior point covariance Σ_j, the predicted point covariance is given as

$$\bar{\Sigma}_j = (I - \Sigma_j J_j^T (J_j \Sigma_j J_j^T + R)^{-1} J_j)\Sigma_j, \quad (4)$$

where $R \in \mathbb{R}^{2\times2}$ is the noise covariance for the image point measurement.

Note that the prior point covariance, Σ_j, is available from the most recent bundle adjustment. The predicted covariance $\bar{\Sigma}_j$ provides an estimate of the covariance for point, p_j, if the observing keyframe was inserted into the bundle adjustment process. Equation 4 can be evaluated rapidly for each point, as the computational bottleneck is the inversion of a 3 by 3 matrix. Although comparison of the covariance matrices provides information on the uncertainty reduction for one point, the covariance representation does not allow for a convenient way to assess the uncertainty reduction across all of the points observed in the multi-keyframe. To that end, we propose evaluation of the uncertainty reduction using the point *entropy*.

Denote the entropy corresponding to the point's prior and predicted covariance as $h_e(\hat{p}_j)$ and $\bar{h}_e(\hat{p}_j)$, respectively. The reduction in entropy of point p_j is given as $\Lambda(p_j) = h_e(\hat{p}_j) - \bar{h}_e(\hat{p}_j)$.

Then, the expected entropy reduction, for all map points measured in multi-keyframe M_i is

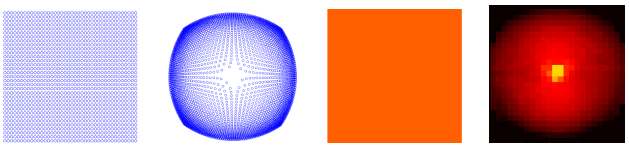
$$\Psi(M_i) = \sum_{K_l \in M_i} \sum_{p \in \tilde{P}_{K_{il}}} \Lambda(p). \tag{5}$$

The multi-keyframe, M_i^*, which maximizes the point entropy reduction: $M_i^* = \underset{M_i \in \Phi}{\operatorname{argmax}}\ \Psi(M_i)$ is selected for insertion in the map through bundle adjustment.

Note that our proposed approach treats each point independently, as the covariance matrix from Equation 4, is only for the parameters of point $\hat{p}_j$, and ignores the correlation the point may have with other points in the map. This independence approximation is suitable for our application, as treating each point independently and summing over all the point entropy reductions provides a lower bound on the overall expected entropy reduction of the map. The mutual information which exists between points [13] will only further reduce the map entropy after the keyframe is inserted and processed with bundle adjustment.

The entropy reduction of a point is also dependent on the camera model, as the point projection Jacobian is dependent on the underlying camera projection equations. It is important to consider the camera model in the formulation, since wide FOV lenses using the Taylor model spatially compress points near the boundaries of the image plane, while a typical pinhole projection preserves the relative spatial distribution of 3D points. This comparison is illustrated in Figures 1(a) and 1(b), which display the image projection of a planar grid of 3D points positioned in front of the camera. As a result of the compression, points projected using the Taylor model, which fall near the boundaries of the image, are less sensitive to perturbations of the 3D point location (see Figures 1(c) and 1(d)). It is evident that the pinhole model maintains uniform sensitivity to point parameter perturbations across the image plane, while the Taylor camera model has reduced sensitivity as the points are projected farther from the image center.

It should be noted that although we derived the proposed keyframe selection approach for points parameterized with Euclidean co-ordinates in the global frame, our method can be formulated for a wide range of point parameterizations, such as spherical, relative, etc.



(a) Pinhole (b) Taylor (c) Jacobian (pinhole) (d) Jacobian (Taylor)

Fig. 1. Comparison of image reprojection sensitivity between pinhole and Taylor camera models, with respect to the x point parameter. Taylor image compression results in reduced sensitivity of the image projection Jacobian at the edges, as seen in (b). Pinhole camera model displays uniform sensitivity in the image reprojection Jacobian, as seen in (a).

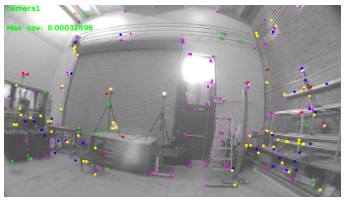
B. Selection based on Point Pixel Flow Discrepancy

Although it is important to strengthen the parameter estimates for the points which already exist in the map, an equally important function for multi-keyframe insertion is to incorporate new map points to track against in the future, particularly when exploring new regions. Feature matches, which exist between the inserted multi-keyframe and the existing multi-keyframes, result in new map points which can used for localization. However, the effectiveness of the newly created points is dependent on the location of the points, as well as the motion of the vehicle. From a tracking perspective, map points should be distributed throughout the image [16] and should be trackable given the current camera motion.

The difficulty in assessing a potentially new map point is that a correspondence is required from an existing keyframe in order to perform the 3D point triangulation. Although it would be possible to develop a heuristic to postulate the effectiveness of an image feature *without* searching for correspondences, performing feature matching in the keyframe evaluation process presents many benefits, such as using the 3D location of the point to predict its motion with respect to a camera fixed observation frame, and detecting when measurements to a feature are occluded due to obstacles. We propose a three-step method to evaluate multi-keyframes from the multi-keyframe buffer, Φ, for potential map points.

Step 1: We first project the map points onto the image frame corresponding to the current tracker position in order to determine how they will flow in the image, given a time horizon and a predicted camera cluster motion. For this derivation, we use a simple constant velocity model, however, the approach can be generalized to more complex motion models.

The flow of the map point, p_i, on the image will begin at image co-ordinates corresponding to the point reprojection, $u_i = \Pi(p_i)$, and will flow according to the Jacobian of the image reprojection for point i, with respect to the cluster state, ω, which is given by $G_i = \frac{\partial \Pi(p_i)}{\partial \omega}$. If the motion of the camera cluster with respect to time is denoted as $\frac{\partial \omega}{\partial t}$, then the change of the point projection with respect to time



(a) Camera Image with Map Points

(b) E-PFP

(c) F-PFP

Fig. 2. (a) Camera image. Red, yellow, green, and blue markers denote existing map points from finest to coarsest image pyramid levels. Magenta markers denote predicted point locations triangulated between current image and existing multi-keyframes. (b) E-PFP and (c) F-PFP use 1m forward motion. Light regions denote areas of high probability, while darker areas indicate low probability.

is $\frac{\partial \Pi(p_i)}{\partial t} = \frac{\partial \Pi(p_i)}{\partial \omega}\frac{\partial \omega}{\partial t}$. Using the constant velocity model over a specified time horizon, Δ_t, the ending image point, $\bar{u}_i$, is

$$\bar{u}_i = \frac{\partial \Pi(p_i)}{\partial t}\Delta_t. \tag{6}$$

Using the pair of start and end points for the reprojection flow $(u_i, \bar{u}_i)$, a discrete probability distribution can be constructed on the image plane, referred to as the *existing point flow PDF (E-PFP)*. The E-PFP captures the probability of *not* measuring existing map points in the image, so areas of high probability denote locations where we would like to *add* new map points. The E-PFP is initialized with a 2D uniform distribution, then straight lines are drawn between the reprojection flow pairs, and the 2D locations on the PDF which intersect the flow lines are assigned probability 0. The flow lines are then dilated and blurred in order to account for motion uncertainty of the camera cluster. Finally, the E-PFP is normalized such that the sum of all probabilities is equal to 1. A depiction of the E-PFP is given in Figure 2(b).

Step 2: We now wish to find the multi-keyframe from the buffer, which will add new image points in high probability areas of the E-PFP, which requires the matching of image features from the candidate keyframe to features located in keyframes in the existing map. Features are binned and only matched against the N closest multi-keyframes for computational efficiency.

To determine how the predicted map points will flow in the image plane, we construct the *future point flow PDF (F-PFP)*. For each candidate multi-keyframe, we perform feature matching and generate the set of predicted map points, $\tilde{P}$, and then compute the pixel flow for the predicted points in $\tilde{P}$ using Equation 6. The F-PFP is also initialized as a discrete uniform distribution, and regions of the predicted points' pixel flow are assigned a probability of 1. Similar to the E-PFP, the flow lines for the predicted points are dilated and blurred, and the PDF is normalized. An example F-PFP is shown in Figure 2(c).

Step 3: We now score each F-PFP from the buffer with the E-PFP. To compare the probability distributions, we compute the relative entropy between the E-PFP and F-PFP using Equation 2. Recall that the if the two PDF's are identical, the relative entropy will be zero. If we denote the E-PFP as α, and the set of all F-PFPs from the buffer of multi-keyframes as $\Gamma = \{\beta_1, \beta_2, \ldots, \beta_k\}$, $k = |\Phi|$, then the F-PFP which minimizes the relative entropy, β^*, is selected as $\beta^* = \underset{\beta \in \Gamma}{\operatorname{argmin}}\ \mathcal{D}(\alpha \| \beta)$.

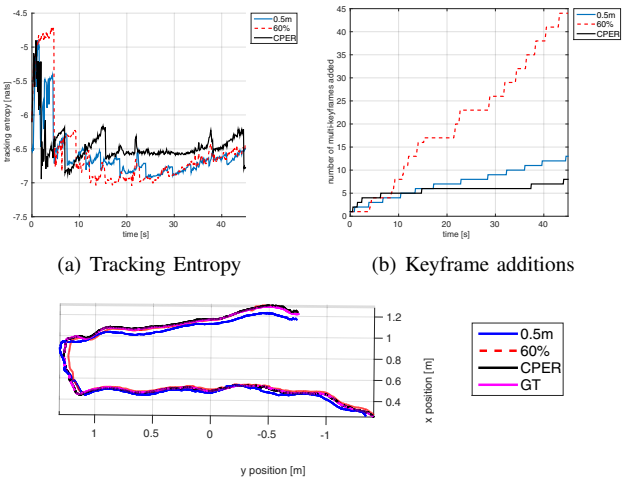
V. Experimental Results

To validate our proposed keyframe selection methods, we performed two sets of experiments. First, we simulate a surveillance task, and compare the CPER approach to other common keyframe selection heuristics based on movement thresholds and map point overlap. Second, we demonstrate the effectiveness of the PPFD approach in exploration scenarios involving occlusions and aggressive camera motion. For both cases, the keyframe selection methods are implemented within the MCPTAM framework. The camera rig was equipped with three rigidly mounted Ximia xiQ cameras fitted with 160 degree wide FOV lenses, with one camera looking forwards, and the others facing off to the left and right sides. Images were captured at 30 Hz, and ground truth of the camera motion was collected using an Optitrack Indoor Positioning System (IPS), which is capable of tracking motion with sub centimeter translational accuracy, and sub degree rotational accuracy.

A. Evaluation of CPER in a Surveillance Task

We simulate a surveillance task by moving the camera rig in an environment with few occlusions, where many features can be re-observed throughout the trajectory. The test trajectory is visualized in Figure 3(c). The proposed CPER approach is compared against a distance threshold heuristic, which adds a keyframe based on the distance between the current tracker position and the existing multi-keyframes, and a point overlap heuristic, which adds a keyframe when the ratio of measured image features to measured map points falls below a threshold. We tested the CPER method to distance threshold of 0.5m, 1m, and 2m, and to point percent overlap threshold of 25%, 50%, and 60%. For the CPER method, the entropy threshold, ϵ, was set to -6.0 nats. The results of the experiment are shown in Figure 3 and summarized in Table I.

In general, all of the tested approaches were able to provide precise localization, due to the small workspace and high visibility of features. However, the CPER approach demonstrated an improvement in average translation error of roughly one order of magnitude, compared to the other methods. The rotation errors are approximately equal for all of the tested methods, except for the 25% threshold, which also performed the least well in terms of translation error. This is likely because the 25% threshold only added 3 keyframes to the map, resulting in few points to track against.



(a) Tracking Entropy (b) Keyframe additions

(c) Camera Trajectories

Fig. 3. (a) Tracking entropy over time. (b) Keyframe additions over time. The CPER method aggressively adds keyframes in the first few seconds of motion in order to rapidly lower the tracking entropy. (c) Recovered camera cluster motion for 0.5m, 60%, CPER and ground truth. 1m, 2m, 25%, and 50% threshold approaches are omitted as they performed similarly.

TABLE I

SUMMARY OF RESULTS FOR SURVEILLANCE MOTION SEQUENCE

	avg. sqr. trans. err [m]	**avg. sqr. rot. err [rad]**	**no. multi-kf**	**no. map points**	**avg. trk. ent. [nats]**
0.5m	0.0024	4.33e-05	13	2431	-6.59
1m	0.0027	5.50e-05	5	1390	-6.22
2m	0.0039	1.75e-04	4	459	-5.73
25%	0.0105	0.0019	**3**	**104**	-4.7
50%	0.0023	1.00e-04	11	1886	-6.11
60%	**7.02e-04**	1.33e-04	44	3657	-6.55
CPER	1.04e-04	**4.12e-05**	8	949	**-6.56**

CPER aggressively added 4 key-frames within the first two seconds of motion to rapidly reduce the tracking entropy, as seen in Figure 3(b). Although the decrease in entropy, seen in Figure 3(a), is also exhibited by the 60% and 0.5m threshold approaches, both achieve the same entropy reduction approximately 3 seconds after CPER. Further, although the 0.5m and 60% threshold achieved comparable accuracy and tracking entropy reduction, each approach added 13 and 44 keyframes respectively, while CPER only added 8. By only adding keyframes when required by the tracker, and selecting keyframes by their contribution to map quality, CPER uses fewer images to produce higher quality maps. Note that image noise and temporary tracking loss of map points will cause point percent methods to rapidly insert keyframes into the map, displayed by the 60% threshold approach in Figure 3(b).

B. Evaluation of PPFD in an Exploration Task

We identify three important situations where the PPFD approach is effective in keyframe selection: exploration, occlusion, and aggressive motions. Results from the three cases are highlighted in Figure 4, while the relative entropy score (RES) over the motion sequences are displayed in Figure 5. Keyframe insertion was triggered after a fixed time horizon to decouple the insertion interval from the selection of a new keyframe using the PPFD method.

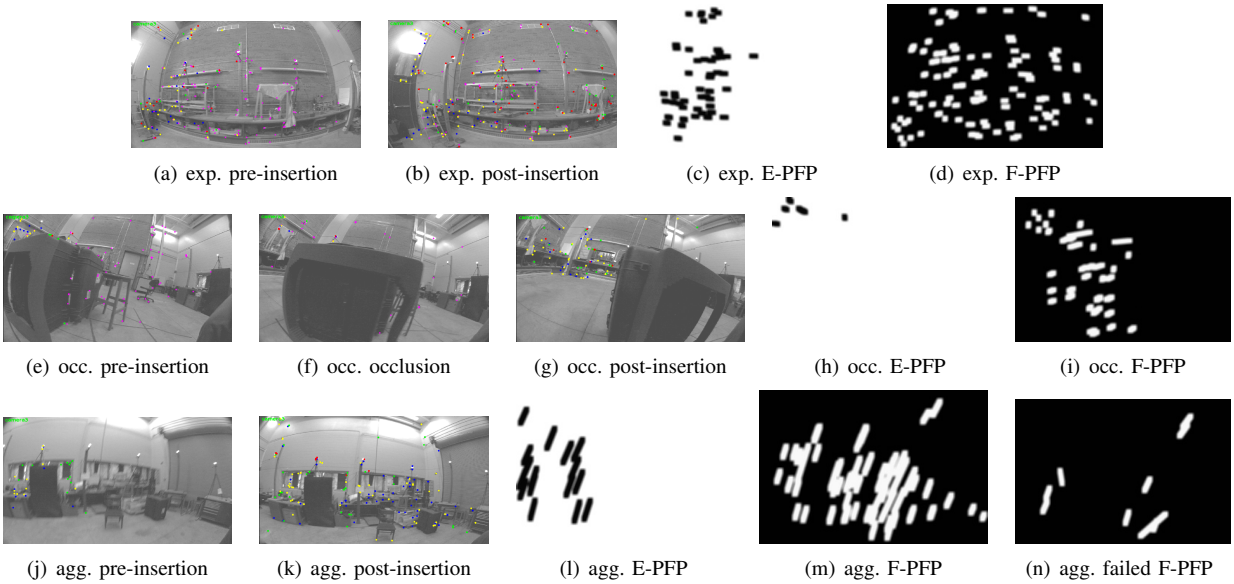
Exploration: Figure 4(a) shows an image with the map points visible immediately after initialization, while Figure 4(b) shows the map points after the keyframe was inserted using PPFD. The respective E-PFP and F-PFP are visualized in Figures 4(c) and 4(d). The E-PFP clearly shows areas in the image frame which are lacking points, and the F-PFP corresponds to the keyframe from the buffer which resulted in the lowest RES, displayed in Figure 5(a).

Occlusion: The map points pre- and post-keyframe insertion are depicted in Figure 4(e) and 4(g), respectively. There are few map points at initialization, and as shown in Figure 4(f), there are few predicted map points due to the occlusion. This behavior is reflected by the RES in Figure 5(b), where the occlusion results in a high RES between keyframes 10 and 85. Finally, once the camera has passed the occlusion, the RES drops sharply at keyframe 90, which is selected as the keyframe for insertion. The respective E-PFP and F-PFP are visualized in Figures 4(h) and 4(i).

Aggressive Motion: Due to the aggressive movement of the camera, the image exhibits motion blur, as seen in Figure 4(j). In this case, the camera was pitched back and forth aggressively, which resulted in a diminished ability to robustly track features. Although the RES, displayed in Figure 5(c), exhibits oscillatory behavior due to poor feature matching, the PPFD method chooses the keyframe from the buffer, which is able to best initialize new map points. In this case, keyframe 10, where the image was momentarily sharp due to a direction change in the rotation, was selected for insertion, resulting in the new map points seen in Figure 4(k). The associated E-PFP and F-PFP are also visualized in Figures 4(l) and 4(m), respectively. Figure 4(n) depicts an F-PFP from keyframe 110, which had a relatively high RES. Note the lower number of features detected on Figure 4(n) when compared to Figure 4(m).

VI. CONCLUSION

This paper demonstrates two approaches for keyframe selection, cumulative point entropy reduction (CPER) and predicted pixel flow discrepancy (PPFD), which systematically evaluate candidate keyframes and selects them based on the expected improvement of the map. We provide experimental results collected with an indoor positioning system which demonstrate that the CPER approach works well in surveillance tasks, as the method provides increased localization accuracy compared to existing methods, while inserting fewer keyframes and map points. We also show that the PPFD approach is able to select keyframes which are especially effective in situations where the camera cluster undergoes exploration, encounters occlusions, and experiences aggressive motions. Our future work involves combining the CPER and PPFD approaches within a unified framework, and verifying our methods in challenging outdoor environments.

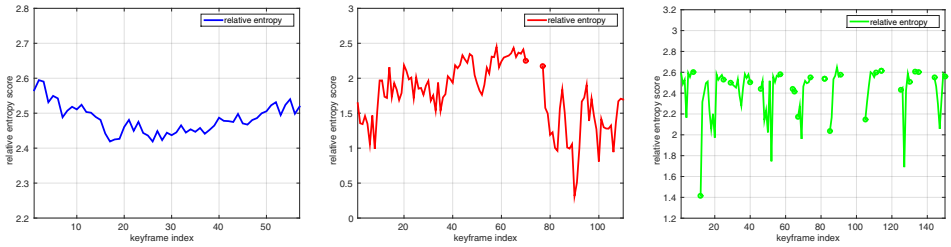


(a) exp. pre-insertion (b) exp. post-insertion (c) exp. E-PFP (d) exp. F-PFP

(e) occ. pre-insertion (f) occ. occlusion (g) occ. post-insertion (h) occ. E-PFP (i) occ. F-PFP

(j) agg. pre-insertion (k) agg. post-insertion (l) agg. E-PFP (m) agg. F-PFP (n) agg. failed F-PFP

Fig. 4. Subfigure sets (a)(b), (e)(f)(g), and(j)(k) show the map points overlayed in the camera images, for the exploration (exp), occlusion (occ), and aggressive (agg) motion cases, respectively. Similarly, sets (c)(d), (h)(i), and (l)(m) visualize the E-PFPs and F-PFPs used to select the inserted keyframe. Finally, (n) depicts the F-PFP of a keyframe from the aggressive motion case, where few feature matches are found due to motion blur.



(a) relative entropy (exploration) (b) relative entropy (occlusion) (c) relative entropy (aggressive motion)

Fig. 5. Plots (a),(b), and (c) show the relative entropy over the sequence for the forward exploration, occlusion, and aggressive motion cases, respectively. Discontinuities in the plots are due to indeterminate relative entropy scores that result when there are no feature matches in the keyframe. Such is especially the case in (c), where rapid motion occurs.

REFERENCES

[1] A. Kim and R. Eustice, "Real-time visual SLAM for autonomous underwater hull inspection using visual saliency," *IEEE Transactions on Robotics*, vol. 29, no. 3, pp. 719–733, 2013.

[2] L. Doitsidis, A. Renzaglia, S. Weiss, E. Kosmatopoulos, D. Scaramuzza, and R. Siegwart, "3D surveillance coverage using maps extracted by a monocular slam algorithm," in *International Conference on Intelligent Robots and Systems (IROS)*, Sept 2011, pp. 1661–1667.

[3] T. Howard, A. Morfopoulos, J. Morrison, Y. Kuwata, C. Villalpando, L. Matthies, and M. McHenry, "Enabling continuous planetary rover navigation through FPGA stereo and visual odometry," in *IEEE Aerospace Conference*, 2012.

[4] A. Das, D. Kumar, A. El Bably, and S. L. Waslander, "Taming the north: Multi-camera parallel tracking and mapping in snow-laden environments," in *Field and Service Robotics (FSR)*, June 2015.

[5] G. Klein and D. Murray, "Parallel tracking and mapping for small AR workspaces," in *IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR)*, Nara, Japan, November 2007, pp. 225–234.

[6] S. Leutenegger, P. T. Furgale, V. Rabaud, M. Chli, K. Konolige, and R. Siegwart, "Keyframe-based visual-inertial SLAM using nonlinear optimization," in *Robotics: Science and Systems (RSS)*, June 2013.

[7] D. Gallup, J.-M. Frahm, P. Mordohai, and M. Pollefeys, "Variable baseline/resolution stereo," in *Computer Vision and Pattern Recognition (CPVR)*. IEEE, June 2008, pp. 1–8.

[8] Z. Dong, G. Zhang, J. Jia, and H. Bao, "Keyframe-based real-time camera tracking," in *International Conference on Computer Vision (ICCV)*. IEEE, Sept. 2009, pp. 1538–1545.

[9] A. Handa, M. Chli, H. Strasdat, and A. Davison, "Scalable active matching," in *Computer Vision and Pattern Recognition (CVPR)*, June 2010, pp. 1546–1553.

[10] M. Chli and A. Davison, "Automatically and efficiently inferring the hierarchical structure of visual maps," in *International Conference on Robotics and Automation (ICRA)*, May 2009, pp. 387–394.

[11] D. Scaramuzza, A. Martinelli, and R. Siegwart, "A flexible technique for accurate omnidirectional camera calibration and structure from motion," in *IEEE International Conference on Computer Vision Systems (ICVS)*. New York, NY: IEEE, Jan. 2006, pp. 45–45.

[12] J. Engel, T. Schöps, and D. Cremers, "LSD-SLAM: Large-scale direct monocular SLAM," in *European Conference on Computer Vision (ECCV)*, Sept. 2014.

[13] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. New York, NY: Wiley-Interscience, 2006.

[14] A. Harmat, M. Trentini, and I. Sharf, "Multi-camera tracking and mapping for unmanned aerial vehicles in unstructured environments," *Journal of Intelligent & Robotic Systems*, pp. 1–27, 2014.

[15] A. Tribou, Michael J. and. Harmat, D. Wang, I. Sharf, and S. L. Waslander, "Multi-camera parallel tracking and mapping with non-overlapping fields of view," *International Journal of Robotics Research: to appear*, pp. 1–43, December 2014.

[16] A. Geiger, J. Ziegler, and C. Stiller, "Stereoscan: Dense 3D reconstruction in real-time," in *Intelligent Vehicles Symposium (IV)*, June 2011.