A Minimum Variance Calibration Algorithm for Pan-Tilt Robotic Cameras in Natural Environments

Dezhen Song[1], Ni Qin[1], and Ken Goldberg[2]

1: CS Department, Texas A&M University, College Station, TX 77843
2: IEOR and EECS Departments, University of California, Berkeley, CA 94720

***Abstract*— A new generation of inexpensive robotic pan-tilt cameras can maintain high-resolution panoramic displays of natural environments. However, the pan-tilt mechanisms are imprecise: small errors can produce large errors in the panoramic display. It is thus important to accurately estimate pan-tilt values. We present a new calibration algorithm that does not rely on calibration markers or fixed orthogonal edges which are rarely available in natural scenes. Our calibration algorithm uses image variance density to optimally estimate camera pan and tilt values by incrementally refining image registration using overlapping images from prior frames. Experiments suggest that the new calibration algorithm can reduce calibration error by 81%. In a companion paper [19], we present a new image registration algorithm based on spherical projection that optimally aligns the resulting frames.**

I. Introduction

Scientific study of wild animals requires continuous observation over a distance. Recent developments in wireless telecommunications facilitate low-bandwidth connectivity to remote sites. A new class of low-cost tele-operated pan-tilt-zoom robotic video cameras allows fast deployment of systems that can provide high-resolution images from a wide field of view in a remote environment. One example is the Panasonic HCM 280 camera with built-in streaming server, 22x zoom motorized optical lens, 350° pan range, and 90° tilt range. However, minor errors in the camera pan and tilt mechanism can produce large errors between nominal camera coverage and actual coverage as illustrated in Figure 1. For example, an error of $0.5°$ in camera tilt position can cause a 41.67% error in coverage when a Panasonic HCM 280 camera operates at its highest zoom.

Recent progress in camera calibration provides the capability to calibrate an outdoor camera under natural lighting without using predefined calibration objects [7]. Those methods utilize linear edges in a rigid scene such as contours of buildings as a calibration reference. Unfortunately, it cannot be directly applied to online calibration problems in a natural environment where linear edges are often not available.

We notice that intrinsic camera parameters such as CCD sensor size, focus length, and skew factor do not change over time. Hence, we assume these parameters are known and focus on calibrating the pan and tilt mechanism. We assume that

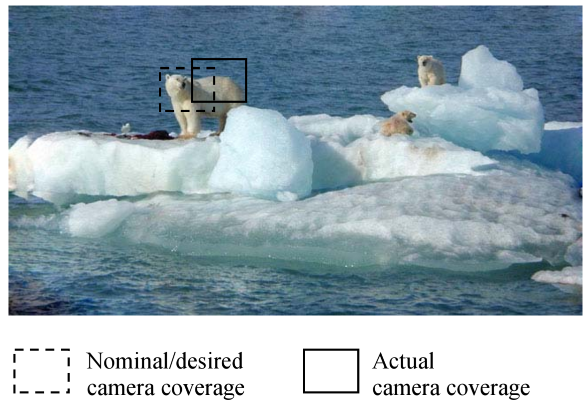

Fig. 1. *Operators' desired camera coverage and the actual camera coverage cannot perfectly match with each other when there is a calibration error. The calibration error causes difficulties when the tele-operated pan-tilt robotic camera is used to track moving animals over a distance.*

pan and tilt potentiometer readings are approximate or have a limited accuracy. We propose a new calibration algorithm that optimally estimates the pan and tilt positions for a new camera frame by incrementally refining the image registration solution for prior overlapping frames in increasing order of location variance density. For k images, our algorithm runs in time $O(k \log k)$. Experiments show that our algorithm can reduce calibration error by 81% if compare with a method that simply selects frames with large overlapping regions.

II. Related work

Camera calibration is used to determine accurate intrinsic parameters (CCD sensor size, skew factor, focus length, and lens distortion) and/or extrinsic parameters (position and orientation of a camera). Our calibration problem focuses on the camera pan and tilt mechanism, which is part of a camera's extrinsic parameters.

The fundamental work on camera calibration is to calibrate a still camera using still calibration objects, which is credited as *photogrammetric calibration*. This is based on a parameterized camera imaging model such as the pinhole perspective projection model [26]. The unique characteristics of the imaging model distinguish camera calibration from robot calibration. Imprecision in camera parameters causes discrepancy between image coordinate systems and the world coordinate system. Sometimes, the discrepancy is caused by the fact that the pinhole model, which is an approximate model itself, cannot

This work was supported in part by the National Science Foundation under IIS-0534848 and IIS-0535218, by Intel Corporation, by Panasonic, by TAMU startup fund, and by UC Berkeley's Center for Information Technology Research in the Interest of Society (CITRIS). For more information please contact dzsong@cs.tamu.edu or goldberg@ieor.berkeley.edu.

accurately model the imaging process [11]. The calibration process can be viewed as model-fitting or parameter identification. Photogrammetric calibration observes a 3D calibration object with a known geometry, usually consisting of several mutually orthogonal planes [3], [8], [9], [15]. This approach is very efficient, but requires carefully designed calibration objects located at 2 or 3 orthogonal planes [26].

Since it is not convenient to accurately set up calibration objects involving 2 or 3 orthogonal planes, a different approach, known as *self-calibration*, does not use any calibration object, but relies on the rigidity of a scene to calibrate calibrate intrinsic camera parameters. It takes a series of images while moving a camera in a static scene. The assumed rigidity of a scene is used to compute camera parameters [5], [6], [13], [14], [16]–[18], [21], [23], [24]. Self-calibration reduces the complexity of setting up the calibration process by assuming the perfect motion accuracy, which can be viewed as the inverse of the problem that we are facing. Because we assume known intrinsic parameters, only a camera's pan and tilt mechanism needs to be calibrated.

Realizing that it is not convenient to use a predefined calibration pattern to calibrate an outdoor camera, Basu and Ravi [2] propose the notion of *active calibration*, which calibrates the image center and focus of a camera using a set of linear edges in a scene. Collins and Tsin [7] further developed this idea, which can estimate both intrinsic and extrinsic camera parameters for pan-tilt-zoom cameras based on how the linear edges change while rotation and zoom operations are performed. Since linear edges are difficult to find in a natural environment, we expand the approach by using the offset generated by imaging alignment techniques to replace linear edges. We address the error propagation problems in multiple imaging alignments by choosing an optimal set of images to calibrate the pan and tilt positions for a newly-captured image.

Sinha and Pollefeys [20] develop automatic calibration algorithms for a pan-tilt-zoom camera with a focus on automatic zoom calibration. Similar to our approach, their method does not require a structured scene or calibration object. They first determine intrinsic camera parameters at the lowest zoom and then increase camera zoom settings to obtain radial distortion parameters. During the calibration, they focus on intrinsic parameter calibration and assume camera pan-tilt is accurate and repeatable. They obtain extrinsic parameters by matching images captured to a pre-constructed panorama. The accuracy of extrinsic parameters depends on the panorama quality, which is sensitive to the number of frames and lighting variations across frames. This paper complements their work by concentrating on extrinsic parameter calibration without assuming an existing panorama.

III. Problem Description

Below, we briefly review our previous work on extrinsic camera calibration algorithms based on image features identified by remote human operators. To eliminate humans in the calibration procedure, we propose to use image alignment to eliminate manual identification of feature points and formulate an online pan-tilt calibration problem. We begin with assumptions.

A. Assumptions

We assume that all images are taken from a fixed camera, which only performs pan and tilt movements. Due to cost and space limitations, the camera's angular potentiometer usually has limited accuracy and may deteriorate over time. Hence these extrinsic parameter readings are inherently approximate and need to be calibrated. We assume that the rate of the error change is slow and hence periodic calibration can compensate for it. We assume that the intrinsic parameters including image resolution, camera focus length, and CCD sensor size are pre-calibrated and known.

B. Nomenclature and Feature-based Calibration

Salient and fixed points in the environment are identified as *calibration feature points*. Figure 2 shows some sample feature points including the center of a clock, a corner of the bookshelf, a power outlet on the wall, and a nail in the wall. For the jth feature point, we can pan and tilt the camera to center a frame j on it. Hence, we also reference it as the jth frame.

Let us define the following variables,

- $X_j^* = (p_j^*, t_j^*)$: true camera position of the jth frame. It remains unknown during the entire calibration process. Sometimes, it is also referred to as an optimal camera position because the calibration problem is an error-minimization problem and the true camera position is the optimal position.
- $\hat{X}_j = (\hat{p}_j, \hat{t}_j)$: the nominal camera position of the jth frame. It is the reading from the camera potentiometer. It is referred to as the nominal camera position because of the errors in the camera potentiometer.
- $X_j = (p_j, t_j)$: the measured camera position of the jth frame. It is the corresponding measured value of the true camera position. The measurement process can be done using pre-defined calibration objects. In our paper, we use image alignment.
- $e_j = X_j - X_j^*$: the measurement error of the jth frame. We assume that the measurement is non-biased, $mean(e_j) = 0$. Therefore, $mean(X_j) = X_j^*$ and $Var(e_j) = Var(X_j) = \begin{pmatrix} \sigma^2(p_j) & 0 \\ 0 & \sigma^2(t_j) \end{pmatrix}$ because pan and tilt are independent variables.
- $\sigma_j^2 = \max\{\sigma^2(p_j), \sigma^2(t_j)\}$, the maximum variance of X_j and e_j. The variance metric is used as the performance metric in the rest of the paper.

A pan-tilt calibration is a two-step operation. Firstly, we need to identify calibration feature points and collect their readings $(X_j, \hat{X}_j)$. Secondly, a calibration model is selected before its parameters are estimated using the collected feature points $(X_j, \hat{X}_j)$. Since the second step has been addressed in photogrammetric calibration literature, we concentrate on the first step.

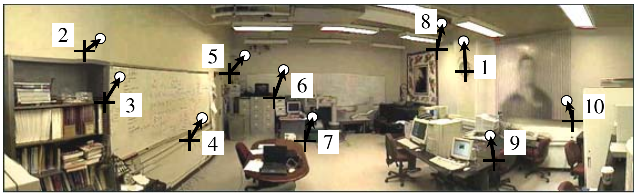

Fig. 2. *Feature point-based pan-tilt-zoom calibration for a robotic camera where features are identified by remote human operators. The measured positions of the frame center are marked as "+", with their nominal positions centered at "○". The arrows indicate error vectors.*

C. Pan-Tilt Calibration Problem Definition

The first step is very difficult to automate in a natural environment because linear edges are rarely available. The measured positions X_j usually need human intervention to be obtained. This is slow and often introduces errors. The results of our previously published work prove that the accuracy of calibration is a linear increasing function of σ_j^2 if the calibration model is a linear model. Hence, the calibration problem becomes how to automatically obtain $X_j = (p_j, t_j)^T$ and also minimize σ_j^2.

Definition 1: Given $\{\hat{X}_j, j = 1, ..., n\}$, find the corresponding $\{X_j, j = 1, ..., n\}$ with the minimum measurement error variance, where n is the total number of feature points.

Since it is an online system, it is desirable to embed the calibration process as a background task. Therefore, the incremental version of the calibration problem is more interesting.

Definition 2: Given $\{\hat{X}_j, X_j, \sigma_j^2, j = 1, ..., n\}$, find the corresponding X_{n+1} with the minimum measurement error variance σ_{n+1}^2 for the newly-captured camera frame $n+1$,

$$\min_{X_{n+1}} \sigma_{n+1}^2(\{\hat{X}_j, X_j, \sigma_j^2, j = 1, ..., n\}). \tag{1}$$

This problem definition actually includes two sub problems: 1) how to obtain X_{n+1} without using manually-selected calibration feature points and 2) how to obtain X_{n+1} with a minimum σ_{n+1}^2.

To get rid of the explicit requirement for calibration feature points, we use image alignment to obtain X_{n+1}. An image alignment problem is a special image registration problem in which all frames share the same optical center and intrinsic camera parameters. A pair-wise image alignment between the newly captured frame $n+1$ and an old frame l, $l < n+1$, outputs the relative offset, $X_{l,n+1}$, between the pan and tilt positions of the two frames. Now the n feature points become pan and tilt positions of the n captured frames. Using $X_{l,n+1}$ and the known pan and tilt position X_l, we can obtain X_{n+1}.

However, we have n existing frames. Each frame has its pan and tilt position X_l and its variance σ_l^2, $l = 1, ..., n$. The relative offset $X_{l,n+1}$ also introduces new variance $\sigma_{l,n+1}^2$. *Our problem becomes to which subset of existing frames should frame $n+1$ be aligned in order to minimize σ_{n+1}^2.* Before we address this problem, let us study how variance has been generated in the image alignment process.

D. Pair-wise Image Alignment

During image alignment, we adopt feature point-based image alignment. Note that the feature points used in image alignment are different from the calibration feature points. The feature points in the image alignment are a noisy set of points detected by a corner detector [12], [28] or a Scale Invariant Feature Transform (SIFT) [4]. Those feature points are detected solely based on pixel values and are not necessarily stable objects in the scene. Therefore, they cannot be directly used for the purpose of calibration. We use feature pixels or pixels to refer to feature points for image alignment in the rest of the paper to avoid confusion.

When aligning a pair of images taken from different pan-tilt settings, perspective projection is required before image matching computations. Details of perspective projection are beyond the scope of this paper and can be found in [27]. After the perspective projection, we adopt the popular Quadratic Chamfer Distance (QCD) [1], [10], [22] to match two images. The QCD measures dissimilarity between two individual feature pixels based on the quadratic distance between the two. Therefore, the Average Matching Error (AME) A of each pixel is a quadratic function in the vicinity of its optimal matching location. The i^{th} pixel in a new frame with its location X_i is described by,

$$A(X_i) = a\|X_i - X_i^*\|_2^2 + b, \tag{2}$$

where X_i^* is the optimal alignment location, and constants a and b are the parameters from the QCD settings. They are the same across all frames. Note that we use the squared Euclidean distance in Equation 2 because of the alignment method adopted. Readers can easily change it to a different distance metric and develop similar algorithms following our techniques.

Define O as the set of overlapping pixels between a pair of intersecting frames. According to Equation 2, the Total Matching Error (TME) T over O becomes,

$$T = \sum_{i \in O} (a\|X_i - X_i^*\|_2^2 + b) \tag{3}$$

$$= |O|a\|X_i - X_i^*\|_2^2 + |O|b. \tag{4}$$

The image alignment is an optimization problem,

$$\arg \min_{\{X_i, i \in O\}} T,$$

subject to the image integrity constraint, which actually reduces the unknown set $\{X_i, i \in O\}$ to the single vector X, which is the relative offset between the pan and tilt positions of the two images. We must find X,

$$T(X) \leq |O|b + \epsilon,$$

where ϵ is the truncation error of the minimization problem. Inserting it into Equation 3, all possible solutions must be inside the ball,

$$\|X - X^*\|_2 \leq \sqrt{\frac{\epsilon}{|O|a}}, \tag{5}$$

where X^* is the optimal solution. Due to the noisy feature pixel set, the AME function is unknown during the problem solving process. Therefore, we cannot directly use X^* deducted from AME as the solution. Any point in the ball with radius $r = \sqrt{\frac{\epsilon}{|O|a}}$ is a possible solution.

How to find a point in the ball depends on the numerical methods adopted. In popular numerical methods for nonlinear optimization like the Gaussian-Newton method, simulated annealing, genetic algorithms, or Random Sample Consensus (RANSAC) [25], the error between the true optimal and the measured solutions, which is the output of the numerical methods, depends on the initial point and truncation error ϵ. A good algorithm chooses its initial point randomly, which defines the alignment error to be a random vector $X = X^* + e$, where e is the alignment error vector. We assume that e has zero mean and variance υ, which means, $mean(X) = X^*$, and $\sigma^2(X) = \sigma^2(e) = \upsilon$.

IV. Algorithms

We use relative offset computed from image alignment to estimate the measured camera position X. Since we have assumed that the error of pair-wise alignment is a random vector with zero mean, the magnitude of the error variance determines the quality of X. We study how error variance gets accumulated and propagated in the alignment process using a simple 1-dimensional example. Based on the analysis, we propose a minimum variance approach to select an optimal set of existing frames to estimate the measured camera pan and tilt position of a newly-captured frame. We begin with the analysis of error variance in the simplest pair-wise alignment operation.

A. Analyzing Error Variance in Pair-wise Alignment

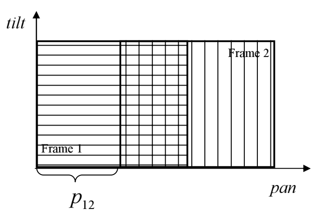

Fig. 3. *An illustration of pair-wise alignment of two equally sized frames with an equal number of pixels.*

Figure 3 illustrates a 1D case where the two frames only have pan displacement. Frame 1 is the reference image in the alignment. Frame 2 is the frame with unknown camera position p_{12}. The pair-wise alignment will determine p_{12} and its variance. Since this is a simple 1-dimensional case, the ball in Equation 5 degrades to a line segment. If we assume the solution p_{12} is uniformly distributed, then its variance υ is

$$\upsilon = \frac{(2r)^2}{12} = \frac{r^2}{3} = \frac{\epsilon}{3|O|a}. \tag{6}$$

Equation 6 is built on two assumptions: 1-dimensional alignment and uniformness in error distribution. A general image alignment case usually involves more than one dimension because images may have both translational and orientational displacements even under our assumptions that intrinsic camera parameters are known. Furthermore, errors are not necessarily uniformly distributed, either.

For a general $d-$dimensional case $X = (x_1, x_2, ..., x_d)^T$ and its alignment error vector $e = (e_1, e_2, ..., e_d)^T = X - X^*$, we have variances of the marginal error distributions along each dimension, $\{\upsilon_{e_1}, \upsilon_{e_2}, ..., \upsilon_{e_d}\}$. We define

$$\upsilon = \max\{\upsilon_{e_1}, \upsilon_{e_2}, ..., \upsilon_{e_d}\} = \max\{\upsilon_{x_1}, \upsilon_{x_2}, ..., \upsilon_{x_d}\},$$

because $\upsilon_{e_j} = \upsilon_{x_j}$, $1 \leq j \leq d$.

Interestingly, though the distribution of the solution point in the ball is unknown, the $d-$dimensional case has a similar format with the 1-dimensional case in Equation 6 with a different constant factor k_d, as summarized in the following theorem.

Theorem 1: Using the AME approximation of image matching function in the vicinity of the optimal solution, the variance of the alignment displacement error is

$$\upsilon = \frac{r^2}{k_d} = \frac{\epsilon}{k_d|O|a}, \tag{7}$$

where $k_d \geq 1$ and d is the problem dimensionality. The exact value of k_d depends on d and the joint probability distribution function of the solution distribution over the ball defined by Equation 5.

Proof: Define the joint probability density function as $f(e_1, e_2, ..., e_d)$, we have

$$\underbrace{\int_{-r}^{r} ... \int_{-r}^{r}}_{d} f(e_1, e_2, ..., e_d) de_1 de_2 ... de_d = 1. \tag{8}$$

Without loss of generality, we assume $\upsilon_{e_1} = \upsilon$. We compute υ_{e_1} in the rest of the proof. Because e_1 has zero mean, we know

$$\upsilon_{e_1} = E(e_1^2) - E^2(e_1) = E(e_1^2).$$

We define,

$$f_1(e_1) = \underbrace{\int_{-r}^{r} ... \int_{-r}^{r}}_{d-1} f(e_1, e_2, ..., e_d) de_2 ... de_d, \tag{9}$$

and

$$F_1(y) = \int_{-r}^{y} f_1(e_1) de_1, \tag{10}$$

as the marginal probability density function and the cumulative probability function for e_1, respectively. Now we are ready to compute υ,

$$\begin{aligned}
\upsilon &= \int_{-r}^{r} e_1^2 f_1(e_1) de_1 \\
&= \int_{-r}^{r} e_1^2 dF_1(e_1) \\
&= e_1^2 F_1(e_1)|_{-r}^{r} - \int_{-r}^{r} 2e_1 F_1(e_1) de_1 \\
&= r^2 - \int_{-r}^{r} 2e_1 F_1(e_1) de_1 \\
&= r^2 - \int_{-r}^{0} 2e_1 F_1(e_1) de_1 - \int_{0}^{r} 2e_1 F_1(e_1) de_1 \\
&= r^2 + \int_{-r}^{0} (-2e_1) F_1(e_1) de_1 - \int_{0}^{r} 2e_1 F_1(e_1) de_1.
\end{aligned}$$

Applying the Second Mean Value Theorem for Integrals, we know $\exists\xi \in [-r, 0]$ and $\exists\zeta \in [0, r]$ such that,

$$\int_{-r}^{0} (-2e_1)F_1(e_1)de_1 = F_1(\xi)\int_{-r}^{0} (-2e_1)de_1 = F_1(\xi)r^2,$$

and

$$\int_{0}^{r} (2e_1)F_1(e_1)de_1 = F_1(\zeta)\int_{0}^{r} (2e_1)de_1 = F_1(\zeta)r^2.$$

Therefore,

$$\upsilon = (1 + F_1(\xi) - F_1(\zeta))r^2,$$

and

$$k_d = 1/(1 + F_1(\xi) - F_1(\zeta)),$$

is the constant. ■

As summarized in Theorem 1, the quality of the solution is determined by how many pixels are involved in the matching, $|O|$, and the image characteristics, a.

B. Analyzing Error Variance in Multi-frame Alignment

Since overall coverage of a pan-tilt-zoom camera is far larger than the coverage of any single camera frame, we have to cascade image alignments to obtain measured camera positions for frames that are relatively far from reference frames. However, such a cascading operation can cause an excessive increase in error variance. We analyze this problem by adding a third frame to the simple 1D case defined in Figure 3.

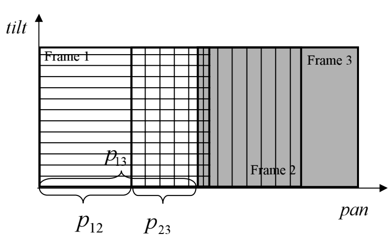

Fig. 4. *Computation of the measured position of a new frame in addition to frame 1 and frame 2 in Figure 3.*

Define m_{12}, $0 \leq m_{12} \leq m$, as the number of overlapping pixels between frame 1 and frame 2. Actually, $m_{12} = |O|$ in Equation 6. Similarly, we have m_{13} and m_{23} introduced by frame 3. As illustrated in Figure 4, define p_{13} and p_{23} as the offset of frame 3 with respect to frame 1 and frame 2, respectively. Recall that p_{12} is the offset of frame 2. Define p^*_{12}, p^*_{13}, and p^*_{23} as the corresponding optimal offsets of p_{12}, p_{13}, and p_{23}, respectively.

Since frame 1 is the reference frame, p_{13} represents the measured camera position of frame 3. We have three different ways to obtain p_{13}:

1) directly compute p_{13} using image alignment since $m_{13} > 0$,
2) compute p_{23} using image alignment, then $p_{13} = p_{12} + p_{23}$, and
3) simultaneously compute both p_{13} and p_{23} using the image alignment with respect to both frame 1 and frame 2.

We are interested in the choice that can minimize the error variance. Below, we analyze the error variance for each of the approaches.

1) Directly compute p_{13}: We can directly apply the result from Equation 6,

$$Var(p_{13}) = \frac{\epsilon}{3m_{13}a}. \tag{11}$$

2) Compute p_{23} first: Similar to Equation 11, we know, $Var(p_{23}) = \frac{\epsilon}{3m_{23}a}$. Since $p_{13} = p_{12} + p_{23}$ and p_{12} and p_{23} are independent random variables, we have,

$$Var(p_{13}) = Var(p_{12}) + Var(p_{23}) = \frac{\epsilon}{3a}\left(\frac{1}{m_{12}} + \frac{1}{m_{23}}\right). \tag{12}$$

3) Simultaneously compute: We can align frame 3 with respect to frame 1 and 2 simultaneously. In this case, the TME in Equation 3 becomes,

$$\begin{aligned} T &= m_{13}\big(a(p_{13} - p^*_{13})^2 + b)\big) \\ &+ m_{23}\big(a(p_{23} - p^*_{23})^2 + b)\big). \end{aligned}$$

Since $p_{23} = p_{13} - p_{12}$ and $p^*_{12} = p^*_{13} - p^*_{23}$,

$$\begin{aligned} T &= (m_{13} + m_{23})\big(a(p_{13} - \frac{m_{13}p^*_{13} + m_{23}(p_{12} + p^*_{23})}{m_{13} + m_{23}})^2\big) \\ &+ \frac{m_{13}m_{23}}{m_{13} + m_{23}}a(p_{12} - p^*_{12})^2 + (m_{13} + m_{23})b. \end{aligned} \tag{13}$$

Using the result from Equation 5, the radius of the ball that covers all possible solutions is $\sqrt{\frac{\epsilon}{(m_{13}+m_{23})a}}$. The variance of the solution for a given p_{12} is,

$$Var(p_{13}|p_{12}) = \frac{\epsilon}{3(m_{13} + m_{23})a}.$$

Equation 13 also tells us the expected solution for a given p_{12} is,

$$E(p_{13}|p_{12}) = \frac{m_{13}p^*_{13} + m_{23}(p_{12} + p^*_{23})}{m_{13} + m_{23}}.$$

From knowledge of conditional variance, we know that

$$Var(p_{13}) = E(Var(p_{13}|p_{12})) + Var(E(p_{13}|p_{12})).$$

Therefore, we can get the variance of the the measured position of frame 3,

$$\begin{aligned} Var(p_{13}) &= \frac{\epsilon}{3(m_{13} + m_{23})a} + \frac{m^2_{23}}{(m_{13} + m_{23})^2}Var(p_{12}) \\ &= \frac{\epsilon}{3(m_{13} + m_{23})a}\left(1 + \frac{m^2_{23}}{m_{12}(m_{13} + m_{23})}\right). \end{aligned} \tag{14}$$

It is desirable to choose p_{12} with the minimum variance. However, depending on the values of m_{12}, m_{13}, and m_{23}, error variances in Equations 11, 12, and 14 do not have a fixed order. Since $\frac{\epsilon}{3a}$ is the common term for all three variance candidates, to find the minimum we must solve the following problem,

$$\min\left\{\frac{1}{m_{13}}, \frac{1}{m_{12}} + \frac{1}{m_{23}}, \frac{1}{m_{13} + m_{23}} + \frac{m^2_{23}}{m_{12}(m_{13} + m_{23})^2}\right\}. \tag{15}$$

Computing the variance for each individual case may be feasible for the simple 1-dimensional case with the small number of frames above. However, we need a more efficient method to find the minimum variance of the measured camera position for a newly-captured frame in a general case, which yields a Minimum Variance Extrinsic Calibration (MVEC) below.

C. Minimum Variance Extrinsic Calibration

Let us consider a general case. Assume that the j^{th} frame enters the system and intersects with a set of existing frames M_j. For the l^{th} frame in M_j, we also know that the number of pixels in frame j intersecting with frame l is m_{jl}. Define X_j and X_l as the vectors that describe the location of image j and image l with respect to the reference image, respectively.

Define X_{jl} and X_{jl}^* as the relative offset and the optimal relative offset between frame j and frame l. Then, the TME formulation of the matching between frame j and all images in set M_j is,

$$T = \sum_{l \in M_j} \big(a m_{jl} \|X_{jl} - X_{jl}^*\|_2^2 + b m_{jl}\big).$$

Since we are looking for the absolute location $X_j = X_l + X_{jl}$, we change the equation above to,

$$T = \sum_{l \in M_j} \big(a m_{jl} \|X_j - X_l - X_{jl}^*\|_2^2 + b m_{jl}\big).$$

Applying the same approach we did for Equation 13, we get

$$E(X_j|\{X_l, l \in M_j\}) = \frac{\sum_{l \in M_j} \big(m_{jl}(X_l + X_{jl}^*)\big)}{\sum_{l \in M_j} m_{jl}}, \tag{16}$$

and

$$Var(X_j|\{X_l, l \in M_j\}) = \frac{\epsilon}{k_d a \sum_{l \in M_j} m_{jl}}.$$

Therefore,

$$\begin{aligned} Var(X_j) &= Var(E(X_j|\{X_l, l \in M_j\})) \\ &+ E(Var(X_j|\{X_l, l \in M_j\})) \\ &= \frac{\sum_{l \in M_j} m_{jl}^2 Var(X_l)}{(\sum_{l \in M_j} m_{jl})^2} \\ &+ \frac{\epsilon}{k_d a \sum_{l \in M_j} m_{jl}}. \end{aligned}$$

From Theorem 1, we know that $Var(X_l) = \frac{\epsilon}{k_d a} w_l$, where w_l was computed when the l^{th} image entered the system. Inserting them into $Var(X_j)$, we get

$$Var(X_j) = \frac{\epsilon}{k_d a}\Big(\frac{1}{\sum_{l \in M_j} m_{jl}} + \frac{\sum_{l \in M_j} m_{jl}^2 w_l}{(\sum_{l \in M_j} m_{jl})^2}\Big). \tag{17}$$

Matching over all overlapping frames may not provide us with the smallest variance. What we want is an optimal set of overlapping frames. If the l^{th} image is not used in the matching, we can simply set $m_{jl} = 0$ in Equation 17 to get the new variance. This defines a minimization problem. Defining $I_l, l \in M_j$ as the image choice variable, we get the following optimization problem,

$$\min F(\{I_l, l \in M_j\}) = \frac{1}{\sum_{l \in M_j} I_l} + \frac{\sum_{l \in M_j} I_l^2 w_l}{(\sum_{l \in M_j} I_l)^2}, \tag{18}$$

subject to

$$\sum_{l \in M_j} I_l \le \bar{m}_j, \tag{19}$$

$$I_l = \{0, m_{jl}\}, \forall l \in M_j, \tag{20}$$

where $\bar{m}_j$ is the maximum limit of the number of pixels involved in the matching problem. The constraint in Equation 19 controls the size of the subsequent matching problem to limit computation time. We solve this optimization problem to derive the optimal set of matching images.

D. Minimum Variance Extrinsic Calibration Algorithm

The optimal solution of Equation 18 yields the minimum variance. However, this is a nonlinear combinatorial problem, which could be very computationally expensive. Though the number of overlapping images $k = |M_j|$ is usually a small number, solving it exhaustively requires time exponential in k.

Looking closer, we observe that when the constraint in Equation 19 is binding,

$$\sum_{l \in M_j} I_l = \bar{m}_j,$$

the objective function in Equation 18 becomes

$$F(\{I_l, l \in M_j\}) = \frac{1}{\bar{m}_j} + \frac{\sum_{l \in M_j} I_l^2 w_l}{(\bar{m}_j)^2}.$$

Then, the minimization problem is simplified as,

$$F' = \min_{\{I_l, l \in M_j\}} \sum_{l \in M_j} I_l^2 w_l, \tag{21}$$

subject to the constraint in Equation 20. The l^{th} candidate-matching image takes m_{jl}-pixel space in total, $\bar{m}_j$ pixels, and contributes $m_{jl}^2 w_l$ to variance if it is selected. The variance per pixel is $m_{jl}^2 w_l / m_{jl} = m_{jl} w_l$. Let us define the candidate solution set as $\hat{M}_j \subseteq M_j$, the sum of the pixels in $\hat{M}_j$ as $s_1 = \sum_{l \in \hat{M}_j} m_{jl}$, and the partial variance sum as $s_2 = \sum_{l \in \hat{M}_j} I_l^2 w_l$. We propose an approach that is based on the order of the variance density and solves the problem for the case that the constraint in Equation 19 is binding. This algorithm takes the images that contribute less variance first and gradually expands the set until it reaches the constraint.

MVEC Algorithm

$\hat{M}_j = \emptyset$, $s_1 = 0$, $s_2 = 0$	$O(1)$
Compute $m_{jl} w_l$, $l \in M_j$,	$O(k)$
Sort $\{m_{jl} w_l, l \in M_j\}$ in ascending order,	$O(k \log k)$
For each l in the ascending sequence of $m_{jl} w_l$,	$O(k)$
If $s_1 + m_{jl} \le \bar{m}_j$,	
$s_1 = s_1 + m_{jl}$, $s_2 = s_2 + m_{jl}^2 w_l$, $\hat{M}_j = \hat{M}_j \cup \{l\}$	
Else	
Break for loop	
End if	
End for	
$F(\hat{M}_j) = \frac{1}{s_1} + \frac{s_2}{s_1^2}$	$O(1)$
Output $\hat{M}_j$ and $F(\hat{M}_j)$	$O(1)$

The algorithm above does not directly offer a solution when $\sum_{l \in M_j} m_{jl} < \bar{m}_j$. This is not a problem, because we can treat $\bar{m}_j$ as a variable to perform a search over it. Recalling F' defined in Equation 21, this new optimization problem is,

$$\min_{\bar{m}_j} \frac{1}{\bar{m}_j} + \frac{F'}{\bar{m}_j^2}, \tag{22}$$

which can be solved straightforwardly by keeping track of the F value in the for loop of the MVEC algorithm. Instead of using the final $F(\hat{M}_j)$, we output the smallest F and its corresponding set of frames. With this modification, we have:

Theorem 2: The MVEC algorithm finds the optimal set of overlapping frames to minimize the variance of the measured pan and tilt location of a newly-captured frame in $O(k \log k)$ time for the frame with k overlapping frames.

V. Experiments and Results

We have installed a Canon VCC3 pan-tilt-zoom camera on the UC Berkeley campus. The camera has a pan range of 180° and a tilt range of 55°. It features an 1/4-inch CCD sensor with a maximum resolution of 768×576. Its horizontal field of view ranges from 4° to 46°. Our PC has a 2.53Ghz Intel Pentium 4 processor with 1GB RAM and an 80GB hard drive.

During the calibration, we direct the camera to visit a set of predefined coordinates. We take 21 320×240-pixel frames. We combine our MVEC algorithm with breadth-first search (BFS) to cover the camera's reachable field of view. The BFS starts with the camera's home position frame, which is also our reference frame. It is node 0 in Figure 5. The BFS incrementally covers all 21 points represented by the 21 nodes in the graph illustrated in Figure 5. The 21 nodes in Figure 5 are numbered according to the order of arrival. Note that nodes 5, 10, 11, 13, 16, and 18 only align with a subset of their neighbors, which confirms our analysis that to align with as many frames as possible does not necessarily minimize the variance.

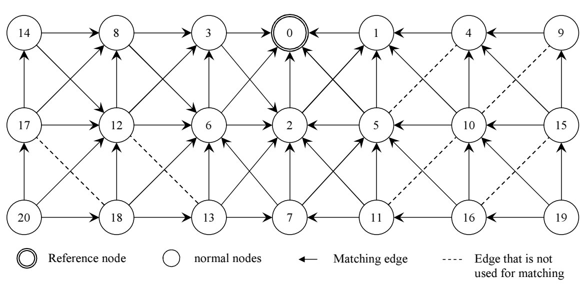

Fig. 5. *Resulting matching sequence from MVEC using 21 frames. Each node represents a frame, and the node numbers correspond to BFS frame capturing order. The distribution of matching edges is determined by image alignment mechanisms. The alignment edges are directional: node a → node b means frame a is captured later and uses the existing frame b for alignment.*

In a second experiment, we compare the calibration accuracy of our MVEC algorithm with that of two other options. The first option is to simply align a newly-captured frame with its recent neighbors, which is called Time-Based Extrinsic Calibration (TBEC). The rationale behind it is that recent neighbors are less vulnerable to the change of environment. The second option is to align the newly-captured frame with the frames with large overlapping regions, which is called Location-Based Extrinsic Calibration (LBEC). The rationale behind it is that large overlaps tend to produce less variance. To ensure a fair comparison, we set the same constraint in Equation 19 across all three options. We select the total number of feature pixels involved as $\bar{m}_j = 5000$. For the TBEC, we rank neighbors according to their arrival time. We add the most recent images to the alignment set until the constraint in Equation 19 is binding. For the LBEC, the only difference is that we rank all neighbors of the new frame according to the size of the overlapping area. For each calibration method, we insert 500 frames into the system as a trial. We repeat each trial 50 times. The data shown in Figure 6 is an average of 50 trials.

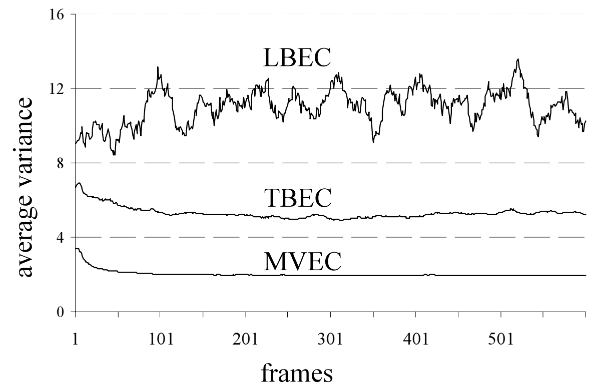

Fig. 6. *A comparison of calibration results using the MVEC algorithm, Location-Based Extrinsic Calibration (LBEC), and Time-Based Extrinsic Calibration (TBEC). The variance unit is $\frac{\epsilon}{k_d a} \times 10^{-3}$.*

Recall that our algorithm selects a subset of frames to align a new frame to minimize the variance of the measured pan and tilt position of the new frame. Therefore, the calibration accuracy is measured using the average variance of the pan and tilt positions of the last 20 frames after the new frame is inserted. Because the variance of a single frame heavily depends on its distance to the reference frame, we use the average of 20 to smooth the location variation in comparison. Since each frame is uniformly, independently, and identically distributed in the camera pan and tilt space, the mean location of the 20 frames is about the same according to the Strong Law of Large Numbers. Although variance usually does not have a unit, Equation 17 suggests that the variance in our system can be measured by constant $\frac{\epsilon}{k_d a}$.

Figure 6 illustrates some interesting results. Both the TBEC algorithm and our MVEC algorithm show a trend of convergence. This is due to the fact that there are not enough pixels to bind the constraint in Equation 19 at the beginning. As more and more frames enter the system, the constraint binds and the average variance converges to a fixed value. It is clear that the MVEC algorithm is more effective in variance reduction. Our data shows that it reduces the variance by 65% on average if compared with TBEC. What surprises us is that the LBEC is actually the worst among the three methods. One big problem is that variance does not converge for the 500 frames inserted. This is because the selection of candidate frames is solely based on the size of the overlapping area, which does not consider the variance of the selected frame. Even after the constraint is binding, a single frame with very large variance can dominate the solution. We know that the variances of initial frames are large. A good method should avoid those frames whenever possible. The TBEC can avoid them over time, but the LBEC fails and hence cannot converge. Our MVEC algorithm reduces variance by 81% on average in comparison to LBEC.

VI. Conclusions and Future Work

We present an algorithm for online extrinsic calibration of a high-resolution robotic camera for remote natural environment observation. Our automatic calibration algorithm utilizes image alignment to obtain the measured camera positions. To control the error introduced by the image matching process, we analyze how errors get accumulated. We then propose a minimum variance extrinsic calibration algorithm. Our algorithm can compute the measured camera position with the minimum error variance in $O(k \log k)$ time for a newly captured frame with k overlapping frames. In the future, we will expand calibration methods to both intrinsic and extrinsic calibration. We will also explore mutual calibration methods for a multiple camera system under a similar variance reduction framework.

Acknowledgments

Thanks are given to Q. Hu, X. Ling, and V. Jan for implementing part of the project. Our thanks to Z. Goodwin, M. Pantaleano, A. Parish, A. Coots, K. Jaehan, N. Amato, D. Volz, T. Ioerger, R. Gutierrez-Osuna, V. Taylor for insightful discussions and feedback.

References

[1] H. G. Barrow, J. M. Tenenbaum, R. C. Bolles, and H. C. Wolf. Parametric correspondence and chamfer matching: Two new techniques fo image matching. In *Proceedings of the Fifth International Joint Conference on Artificial Intelligence*, pages 375–376, 1977.

[2] A. Basu and K. Ravi. Active camera calibration using pan, tilt and roll. *IEEE Transactions on Systems, Man, and Cybernetics*, 27(3):559–566, June 1997.

[3] D.C. Brown. Close-range camera calibration. *Photogrammetric Engineering*, 37(8):855–866, 1971.

[4] M. Brown and D. Lowe. Recognising panoramas. In *In Proceedings of IEEE International Conference on Computer Vision (to appear). IEEE Computer Society, 2003. 64*, 2003.

[5] X. Chen, J. Davis, and P. Slusallek. Wide area camera calibration using virtual calibration objects. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 520–527, June 2000.

[6] T.A. Clarke and J.G. Fryer. The development of camera calibration methods and models. *Photogrammetric Record*, 16(9):51–66, 1998.

[7] R. T. Collins and Y. Tsin. Calibration of an outdoor active camera. Technical report, CMU-RI-TR-98-36, The Robotics Institute, Carnegie Mellon University, 1998.

[8] W. Faig. Calibration of close-range photogrammetry systems: Mathematical formulation. *Photogrammetric Engineering and Remote Sensing*, 41(12):1479–1486, 1975.

[9] O. Faugeras, editor. *Three-dimensional computer vision: a geometric viewpoint*. MIT Press, 1993.

[10] Céline Fouard and Grégoire Malandain. 3-d chamfer distances and norms in anisotropic grids. *Image and Vision Computing*, 23(2):143–158, February 2005.

[11] M. D. Grossberg and S. K. Nayar. A general imaging model and a method for finding its parameters. In *Proceedings of the Eighth IEEE International Conference on Computer Vision*, number 2, pages 108–115, July 2001.

[12] C. J. Harris and M. Stephens. A combined corner and edge detector. In *In Proc. 4th Alvey Vision Conference, Manchester*, pages 147–151, 1988.

[13] Q. Ji and Y. Zhang. Camera calibration with genetic algorithms. *IEEE Transactions on Systems, Man, and Cybernetics, Part A:Systems and Humans*, 31(2):120–130, 2001.

[14] R.K. Lenz and R.Y. Tsai. Techniques for calibration of the scale factor and image center for high accuracy 3-d machine vision metrology. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10(5):713 –720, 1988.

[15] D. Liebowitz and A. Zisserman. Metric rectification for perspective images of planes. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 482–488, 1998.

[16] S.J. Maybank and O. D. Faugeras. Theory of self-calibration of a moving camera. *The International Journal of Computer Vision*, 8(2):123–151, 1992.

[17] K. Nakano, M. Okutomi, and Y. Hasegawak. Camera calibration with precise extraction of feature points using projective transformation. In *IEEE International Conference on Robotics and Automation (ICRA'02)*, volume 3, pages 2532 –2538, May 2002.

[18] M. Personnaz and R. Horaud. Camera calibration: estimation, validation and software. Technical report, INRIA Rhone Alpes No RT-0258, Oct. 2002.

[19] Ni. Qin, D. Song, and K. Goldberg. Aligning windows of live video from an imprecise pan-tilt-zoom robotic camera into a remote panoramic display. In *IEEE International Conference on Robotics and Automation (ICRA), Orlando, Florida*, May 2006.

[20] S. Sinha and M. Pollefeys. Towards calibrating a pan-tilt-zoom cameras network. In *OMNIVIS 2004, ECCV Conference Workshop, Zofin Palace, Slovansky ostrov, Prague 1, Czech Republic*, May 2004.

[21] P. Sturm and S. Maybankt. On plane-based camera calibration: A general algorithm, singularities, applications. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 432–437, 1999.

[22] K. Toyama and A. Blake. Probabilistic tracking in a metric space. In *The Proceedings of the Eighth IEEE International Conference on Computer Vision (ICCV)*, pages 50–57, July 2001.

[23] R.Y. Tsai. An efficient and accurate camera calibration technique for 3d machine vision. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 364–374, 1986.

[24] R.Y. Tsai. A versatile camera calibration technique for high-accuracy 3d machine vision metrology using off-the-shelf tv cameras and lenses. *IEEE Journal of Robotics and Automation*, 3(4):323–344, 1987.

[25] W. Zhang, J. Kosecka, and F. Li. Mosaics construction from a sparse set of views. In *Proceedings of First International Symposium on 3D Data Processing Visualization and Transmission*, pages 177–180, June 2002.

[26] Z. Zhang. A flexible new technique for camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(11):1330–1334, 2000.

[27] B. Zitova and J. Flusser. Image registration methods: a survey. *Image and Vision Computing*, 21(2003):977–1000, June 2003.

[28] M. Zuliani, C. Kenney, and B. S. Manjunath. A mathematical comparison of point detectors. In *Second IEEE Image and Video Registration Workshop (IVR)*, Jun 2004.