Video-based, Real-Time Multi View Stereo

George Vogiatzis[a], Carlos Hernández[b]

[a]*Computer Science, Aston University, Birmingham, B4 7ET, UK*
[b]*Google, Seattle, WA 98103, US*

Abstract

We investigate the problem of obtaining a dense reconstruction in real-time, from a live video stream. In recent years, Multi-view stereo (MVS) has received considerable attention and a number of methods have been proposed. However, most methods operate under the assumption of a relatively sparse set of still images as input and unlimited computation time. Video based MVS has received less attention despite the fact that video sequences offer significant benefits in terms of usability of MVS systems. In this paper we propose a novel video based MVS algorithm that is suitable for real-time, interactive 3d modeling with a hand-held camera. The key idea is a per-pixel, probabilistic depth estimation scheme that updates posterior depth distributions with every new frame. The current implementation is capable of updating 15 million distributions per second. We evaluate the proposed method against the state-of-the-art real-time MVS method and show improvement in terms of accuracy.

1. Introduction

In the last few years, binocular and multi-view stereo (MVS) research has reached a certain level of maturity where high quality reconstruction results can readily be obtained for a variety of scenes [23, 26]. However, the possibility of applying MVS methods to video streams has received less attention. There are several reasons why a video based, real-time MVS system is an interesting alternative to still image systems. For small scale reconstructions, video acquisition can potentially be faster and more user-friendly than acquiring still pictures *e.g.* rapid prototyping or industrial modeling. Similarly, for very large scene reconstructions like city-wide 3d models, where large quantities of data are required, video can offer an affordable way of capturing a massive amount of data in a user-friendly manner. Since large scale reconstruction algorithms are very data hungry, it is no surprise that the two main paradigms used to feed them are either video [6, 22] or community photo collections [1, 21, 24]. Photo collections have the big advantage of being constantly updated. The main disadvantage is that, for the moment, only touristic places such as the Colosseum in Rome or the city of Dubrovnik

Email addresses: g.vogiatzis@aston.ac.uk (George Vogiatzis), carloshernandez@google.com (Carlos Hernández)

have enough photographs to achieve high quality reconstructions [1]. Video on the other hand can be used to reconstruct any scene of interest on demand.

From an algorithmic point of view, video has several characteristics that distinguish it from still image capture. Firstly, the quality of data is typically lower than that of still images in terms of image resolution, motion blur or compression artifacts. Secondly, the quantity of data is orders of magnitude bigger than a still image sequence. Most of the top-performing MVS techniques would not cope in terms of memory and computation time with a relatively small video sequence. The large amounts of data however is also an advantage because it resolves many of the ambiguities inherent in MVS that arise from repeated texture, texture-less regions or occlusion. This is in contrast to the traditional approach of addressing ambiguities in MVS through computationally expensive regularization. Finally, the baseline between successive frames is small which means that image flow is limited to a few pixels between successive frames. The search for correspondences is therefore easier because there are less image locations to search.

In this paper we present an algorithm that exploits the advantages of video input whilst also being resilient to the challenges it poses. The system we describe maintains a large set of candidate 3d features ($\sim 250,000$ of them) at any given point in time. It has an estimate of their 3d position that is improved with every incoming frame. When confidence in this estimate is sufficiently high, the candidate 3d feature is consolidated into a 3d point and leaves the candidate pool while a new one is introduced in its place.

The key requirements of this strategy are:

1. a **sequential update** scheme to obtain intermediate 3d estimates as video frames are acquired,
2. a **precision measure** to assert the accuracy of the 3d estimates,
3. an **outlier rejection** scheme to reliably reject outliers that arise in MVS due to occlusion, lack of texture, etc,
4. **efficiency** both in terms of memory and computation time.

An elegant framework that satisfies the first three requirements is probabilistic inference [16]. In this paper we propose a novel, parametric, Bayesian approximation to the MVS problem that complies with all of the above requirements, including being extremely fast to compute and having a very low memory footprint (each 3d feature uses 9 floats to model the position plus 25 bytes for the reference image patch). The main contributions of this paper are:

1. a probabilistic treatment of occlusion robust photo-consistency [13],
2. a parametric approximation to the full probabilistic inference problem that makes the real-time Bayesian MVS problem tractable,
3. a video based MVS system that is shown to process video input in real-time (60Hz) while providing intermediate reconstruction results as user feedback.

2. Previous work

This paper is primarily related to MVS literature but also to real-time pose and scene reconstruction from video. We start by referring to the MVS evaluation by [23]. Looking at the top performers in that evaluation, we can distinguish two main trends:

region growing methods [8, 11, 12, 20] and occlusion-robust photo-consistency methods [3, 5, 10, 13, 18, 27]. The best performing region growing method [8] uses a combination of photo-consistency based patch fitting, growing and filtering in order to reconstruct the scene of interest. This approach is successful since plane-based photo-consistency performs very well on low textured regions or sparse data sets. However it is not obvious how the patch growing step could be transferred to a video setting. This is because by the time a patch has been optimized and new patches must be generated in its vicinity the camera will have already moved away from that region. The algorithm would therefore have to keep previous images in memory which is not feasible for long sequences at 30-60fps. Alternatively the system would have to wait until the camera revisits the patches which makes it difficult to use in camera drive-through scenarios (e.g. a car-mounted MVS system reconstructing parts of a city).

Occlusion-robust photo-consistency methods provide a very simple pipeline using off-the-shelf algorithms such as dense stereo and 3d segmentation algorithms. However, they rely on a much simpler window-based photo-consistency, less robust to sparse images and lack of texture. The top performer in this group [13] estimates depth by histogram voting of local maxima of a photo-consistency measure. In a real-time video setting however, depth estimation using histograms presents the following difficulties: (a) As new frames are acquired and the histogram is updated with new local maxima, it is not clear how to measure confidence in the current depth estimate. (b) Estimation accuracy depends on bin size. (c) Histograms tend to be memory intensive. Our method is inspired by this second group of methods and proposes a probabilistic interpretation of occlusion-robust photo-consistency. We derive a parametric approximation to the posterior depth distribution which overcomes the difficulties of the histogram approach: (a) The probabilistic framework offers confidence measures for the estimates computed while (b) estimates are not quantized. Finally (c) our representation of the depth posterior has a low memory footprint.

Our work is also related to real-time urban reconstruction methods [6, 22]. While [6] assumes a very simple shape model for the buildings, the method of [22] could be used to reconstruct general 3d scenes. The main differences with their approach are twofold: robustness to camera motion and improved accuracy. The core of their algorithm is based on producing a dense stereo depth-map every 0.5 seconds using the frames captured during that time. The depth-map is then fused in 3d with the previously generated structure resolving any inconsistencies that may arise. This system works very well for car-mounted cameras where the motion of the camera is smooth and with slow-varying speed. If the baseline of the cameras is too small (the car stops) or too big (the car is too fast), the system just drops those frames. This makes their algorithm less suitable for hand-held interactive MVS, our goal, where the camera motion is generally not smooth. In contrast, our formulation is independent of the type of camera motion. Each bit of the geometry is only generated whenever a certain degree of confidence and 3d accuracy is reached. This means that a given part of the geometry with good focus and enough baseline could be generated in just a few tenths of a second while other parts, with occlusions or with shaky camera motion may take longer.

This paper also shares similarities with visual SLAM [7] in the way we sequentially update a pool of 3d features. The main difference is that we aim to maximize the production of 3d features and we assume the camera pose estimation is given so we

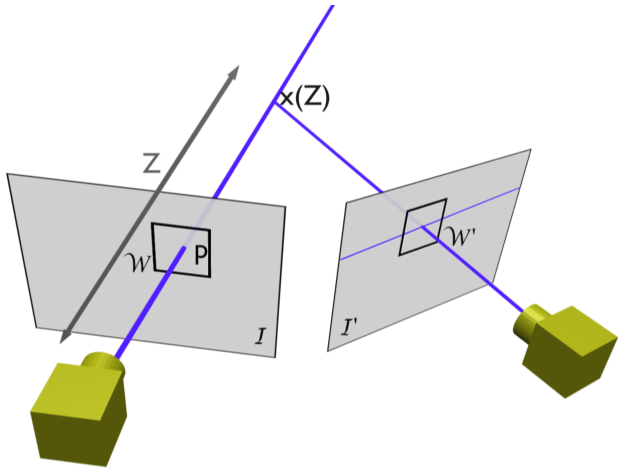


Figure 1: **Searching for a match along an optic ray.** For a given pixel p we wish to find the depth Z along the optic ray through p such that the 3d point $\mathbf{x}(Z)$ projects to similar image regions in images $\mathcal{I}$ and $\mathcal{I}'$. We can measure this similarity by computing a matching score between the two image patches $\mathcal{W}$ and $\mathcal{W}'$.

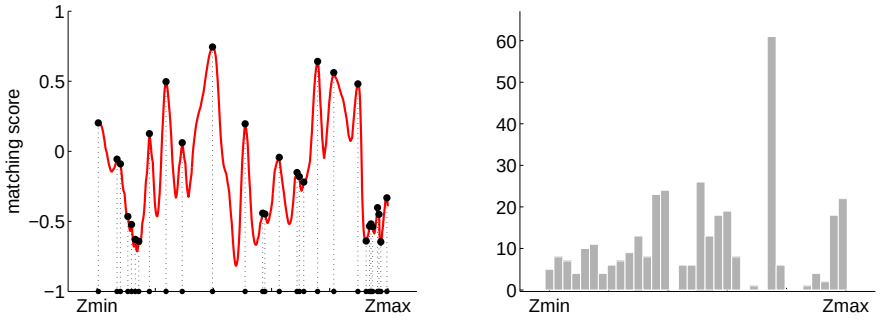
can exploit epipolar geometry.

Finally, a probabilistic approach to MVS has already been proposed in a number of papers [2, 4, 9, 14, 25]. However, while these methods model occlusion explicitly, our approach assumes probabilistic independence of the depth of different pixels and occlusion is implicitly modeled as another source of noise. The independence assumption is key in order to make the real-time MVS problem tractable and we believe that our results justify it in practice.

3. Probabilistic depth sensor

Let p be a pixel of image $\mathcal{I}$ that has been calibrated for pose and internal camera parameters. For a particular depth value Z one can obtain the corresponding 3d point $\mathbf{x}(Z)$ that is located Z units away from $\mathcal{I}$, along the optic ray of p (see figure 1). Let $\mathcal{I}'$ be another calibrated image acquired from a nearby viewpoint while $\mathcal{W}$ and $\mathcal{W}'$ denote two square patches centered on the projection of 3d point $\mathbf{x}(Z)$ onto $\mathcal{I}$ and $\mathcal{I}'$ respectively. We can evaluate the photo-consistency [19] at 3d location $\mathbf{x}(Z)$ using Normalized Cross Correlation (NCC). Figure 2 shows a plot of NCC scores for a pixel across depth as well as a histogram of the local maxima of these curves for 60 neighboring images.

We observe that the histogram is concentrated on a single mode corresponding to the true depth with a uniform component that corresponds to occlusion, image warping, repetitive texture etc. This picture suggests a probability distribution that is a mixture between a good measurement and a bad measurement model. The good measurement model places nearly all its probability mass around the correct depth location while the



(a) (b)

Figure 2: **Depth estimation with NCC maxima.** (a) NCC score across depth along optic ray. The black dots correspond to local maxima. (b) Histogram of local maxima for 60 neighboring images. Local maxima are either generated in the vicinity of the true depth or are uniformly generated across the depth range.

bad measurement model uniformly distributes its mass in all possible depth locations. The next section explains this in more detail.

We view the local maxima $x_1, \ldots, x_N$ as a set of noisy measurements coming from a depth sensor. We model the sensor as a distribution that mixes a good measurement model with a bad one as is common in robust sensor fusion problems (e.g. chapter 21 of [16]). Our sensor can produce two types of measurement with a probability π and $1 - \pi$ respectively: (1) a good measurement that is normally distributed around the correct depth Z or (2) an outlier measurement that is uniformly selected from the interval $[Z_{min}, Z_{max}]$. The limits Z_{min} and Z_{max} can be determined by some prior knowledge of the scene geometry. The object of interest is guaranteed to be entirely contained between Z_{min} and Z_{max}. The following $Gaussian + Uniform$ mixture model describes the probability distribution of the n-th measurement given the correct depth location Z and the inlier probability π

$$p\left(x_n|Z,\pi\right) = \pi N\left(x_n|Z,\tau_n^2\right) + \left(1-\pi\right) U\left(x_n|Z_{min},Z_{max}\right). \tag{1}$$

The variance of a good measurement τ_n^2 can be obtained from the relative position of the cameras at frame $\mathcal{I}$ and $\mathcal{I}'$ that produced the measurement. This is because we assume that the measurement x_n has a fixed variance of one pixel when projected in $\mathcal{I}'$. We then back-project this variance in 3d space to compute the variance of the measurement in distance units.

3.1. Bayesian inference

The likelihood introduced in (1) is a typical mixture model and, as such, its parameters could be estimated from the data $x_1, \ldots, x_N$ in a maximum likelihood framework using Expectation Maximization. However, it is crucial to have a measure of confidence in our depth estimate as it can be used to inform the system when enough measurements have been collected as well as to detect when the estimation has failed. This is not offered by a maximum likelihood approach. Also, in our experiments EM

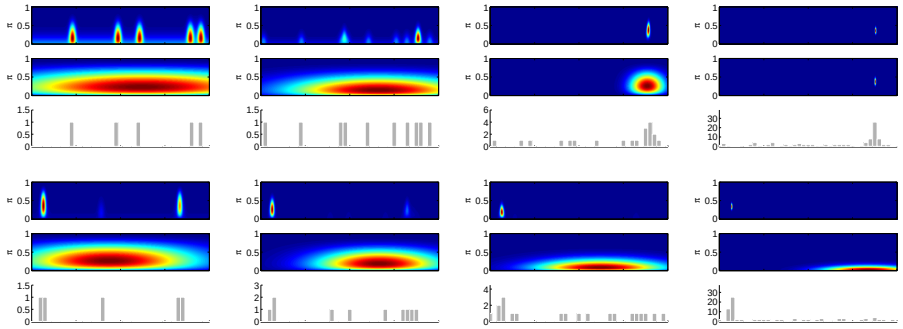


Figure 3: **Non-parametric vs. parametric modeling of posterior distribution.** The first row shows the posterior distribution that is modeled non-parametrically as a 2d histogram. The four columns represent four time instances (after 5, 10, 20 and 100 updates). The second row shows the evolution of our parametric $Gaussian \times Beta$ approximation. Even though our model cannot capture the multi-modal nature of the true posterior, after a few iterations it converges to the same point estimate. The third row shows the histogram of measurements that have been seen by the system in each time instance. The last three rows show one of the few cases where the parametric model cannot follow the non-parametric one. When this happens it can be detected because the parametric posterior predicts a very low inlier ratio. We can therefore safely discard it. The x axis denotes depth along optic ray for all images.

was trapped in a local optimum for a significant percentage of cases. We therefore opt for a Bayesian approach where we define a prior over depth and inlier ratio and then calculate the posterior distribution given all measurements. The estimated depth is then the maximum of this posterior distribution while its shape (through 2nd order moments) determines the confidence in our estimation.

Assuming all the measurements $x_1, \ldots, x_N$ are independent, the posterior has the form

$$p(Z, \pi | x_1 \ldots x_N) \propto p(Z, \pi) \prod_n p\left(x_n | Z, \pi\right). \tag{2}$$

where $p(Z, \pi)$ is our prior on depth and inlier ratio. Figure 3 (top row) shows some snapshots from the evolution of $p(Z, \pi | x_1 \ldots x_n)$ as measurements are collected. The prior is assumed to be uniform and the distribution is modeled using a dense 2d histogram. The posterior converges to the correct values for Z and π. In an experiment described in section 5 and summarized in table 1 we show how this probabilistic formulation outperforms the histogram voting approach used in [13, 27].

However modeling the posterior with a full 2d histogram for each pixel is impractical due to memory and computation limitations. Our system maintains 250,000 seeds at any given time. A reasonable 2d histogram should be quantized with at least 500 values for depth and 100 values for the inlier ratio. To keep these histograms in memory we would need to store 12.5 billion floats (which is non-trivial). Furthermore we found that even with a GPU implementation it was not possible to perform updates of the full 2d histograms in real-time. Our approach is to use a parametric approximation to the posterior as outlined next.

3.2. Parametric approximation to the posterior

When the seed corresponds to a well-textured unoccluded pixel, the histogram of depth observations has a single mode (see figure 5 a). This motivates the use of a uni-modal parametric posterior. A good approximation to the depth posterior (4) is the product of a Gaussian for the depth with a Beta distribution for the inlier ratio. In the supplementary material we provide a variational argument for this form. In particular we show how it has the smallest Kullback Leibler divergence from the true posterior out of a wide set of possible approximation distributions that share a weak factorization property. We therefore define our approximation to the posterior of Eq. (2) as

$$q(Z, \pi | a_n, b_n, \mu_n, \sigma_n) := Beta\left(\pi | a_n, b_n\right) N\left(Z | \mu_n, \sigma_n^2\right). \tag{3}$$

In (3), a_n and b_n can be thought of as probabilistic counters of how many inlier and outlier measurements have occurred during the lifetime of the seed. The other two, μ_n and σ_n^2, represent the mean and variance of our Gaussian depth estimate. Now, if $q(Z, \pi | a_{n-1}, b_{n-1}, \mu_{n-1}, \sigma_{n-1}^2)$ was the true posterior after $n-1$ measurements, the new posterior after observing x_n would have the form

$$C \times p\left(x_n | Z, \pi\right) q(Z, \pi | a_{n-1}, b_{n-1}, \mu_{n-1}, \sigma_{n-1}^2) \tag{4}$$

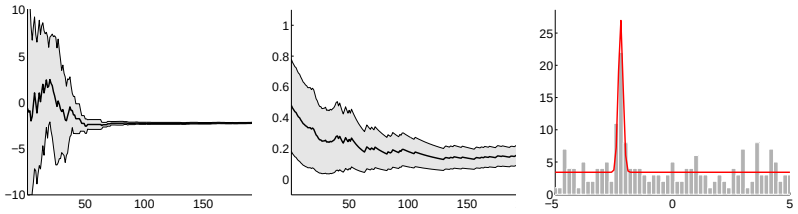
for some constant C. This distribution is no longer of the form $Gaussian \times Beta$ but we can approximate it using moment matching. We therefore define the new parameters $a_n, b_n, \mu_n, \sigma_n^2$ such that the product in (4) and our approximation to the true posterior $q(Z, \pi | a_n, b_n, \mu_n, \sigma_n^2)$ share the same first and second order moments for Z and π. This update is straightforward to calculate analytically but we refer the reader to the supplementary material for the actual formulae. The second and fifth row of figure 3 show the parametric approximation to the posterior as it evolves through the Bayesian updates. Even though our approximation is uni-modal while the true posterior is not, it is nearly always able to converge to the same values of Z and π. In the few cases where it is not able to converge (fifth row of figure 3), the distribution gives high probability to a very low inlier ratio. When this happens, we can be confident that the estimation has failed and we can disregard the results.

Figure 4 shows a typical evolution of this Bayesian update with the parametric approximation. The estimates of Z and π (Fig. 4 a and b) converge to the correct values as can be seen by superimposing the measurement histogram with the marginalized measurement posterior $p\left(x | x_1, \ldots, x_n\right)$ (Fig. 4 c).

It is important to note that the success or failure of the estimation problem also depends upon the quality of the reference patch $\mathcal{W}$ that stays fixed throughout the evolution of the sensor's depth posterior. If that patch is well textured and visible in the subsequent frames the estimation is typically successful. If on the other hand the pixel is either untextured or becomes occluded in the subsequent frames then the estimation fails. Crucially, these failure cases can be detected because the estimated inlier ratio is very low. Such cases are shown in figures 5 b and 5 c.

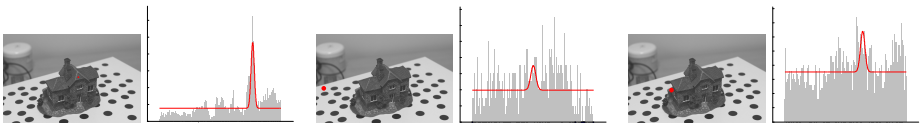
4. System details

One of the aims of this paper is to evaluate the feasibility and usability of a video based multi-view stereo algorithm. In this section we describe a real-time implemen-



(a) (b) (c)

Figure 4: **Parametric update evolution** The first two plots show the evolution of the depth estimate Z and the inlier probability π. We show the mean $\pm$two standard deviations. The third plot shows the measurement histogram superimposed with the measurement posterior $p\left(x|x_1,\ldots,x_n\right)$. Both the mean and the outlier level have been correctly captured.



(a) $\pi = 14.5\%$ (b) $\pi = 3.7\%$ (c) $\pi = 3.4\%$

Figure 5: **Three types of pixel sensors** These figures show the measurement histograms and the superimposed measurement posterior $p\left(x|x_1,\ldots,x_m\right)$ for three types of pixel sensor. In (a) the pixel is a well-textured point on the object. In (b) the pixel corresponds to a completely untextured white point on the ground. In (c) the pixel corresponds to a point that will get occluded within the next few frames. The estimated inlier ratio is shown in the three cases. The two pathological cases, (b) and (c), can be identified from their low inlier ratio.

tation of a reconstruction system based on the ideas described previously. The system is a CUDA based, mixed CPU/GPU implementation. It can update 250,000 seeds, 60 times per second, running on an Intel XEON 2.33GHz with an NVIDIA GTX 260. We also have a portable implementation on a dual-core laptop with an NVIDIA GTX 280M chipset. The portable version updates 250,000 seeds at 30 times per second.

4.1. Camera pose estimation

Recently, there have been major advances in visual SLAM techniques. References to recent work can be found in [7, 17]. These methods can track the 6DOF motion of a camera from point and line features in a video sequence of a rigid scene. In this paper however our aim is to evaluate the dense 3d structure estimation independently from any inaccuracies in a SLAM based camera tracker. To achieve this we chose a simple but very accurate template based camera tracking method using [28]. This technique which includes a per frame bundle-adjustment stage obtains reprojection errors of less than 0.05 pixels.

4.2. Evolution of seeds

The system performs sequential inference of the depth of various pixels in the video sequence. A *seed* corresponds to a particular pixel p whose depth we aim to estimate. Due to memory and computation limitations we can maintain a fixed number of seeds throughout the process. Each seed is associated with a set of parameter values $\left(a_n, b_n, \mu_n, \sigma_n^2, \mathcal{W}\right)$. The first four parameters that evolve during the lifetime of the seed, describe our posterior given the first n observations according to (3). With each seed we also store a reference image patch $\mathcal{W}$ around the pixel location of the seed on the reference image, $\mathcal{I}$. This patch remains constant and is used to compare against target images and obtain depth measurements. When a seed is created we set $a_0 = 10$ and $b_0 = 10$. This corresponds to a prior for the inlier ratio centered on 0.5 and with a standard deviation of approximately 0.1. The depth parameters μ_n and σ_n^2 are set such that 99% of the prior probability mass is between some preset Z_{min} and Z_{max}. These limits define a bounding volume which is known to contain the object of interest.

During the lifetime of a seed we obtain depth measurements by evaluating NCC between the stored patch $\mathcal{W}$ and patches $\mathcal{W}'$ on the epipolar line on the current frame $\mathcal{I}'$ (see fig. 1). Ideally we would like to search the entire epipolar line for local maxima in NCC score but this is not feasible computationally with ordinary hardware. Instead, we exploit the small inter-frame motion by only searching within a radius of w pixels away from the projection of the prior mean μ_n. This violates the independence assumption of Eq. (2) because previous measurements will now dictate the search region for new measurements. In spite of this the approximation works well in practice. In cases when the true depth falls outside this search window of the epipolar line the seed will be producing erroneous depth measurements. We rely on the inlier ratio estimation to detect that the measurements coming from this seed are outliers. The seed will subsequently be discarded as outlined in the next section. In the experiments shown in this paper w is set to 3 pixels for our 2 million pixel camera. In the case when no local maximum is detected, we penalize the seed by setting $b_{n+1} := b_n + 1$. This has the same effect as observing a depth measurement which was known with certainty to be an outlier.

Algorithm 1 The video based MVS algorithm.

$M :=$ the maximum number of seeds that can be maintained in memory.
$S :=$ the current number of seeds in the system
For each new frame $\mathcal{I}$

1. If $S < M$ generate $M - S$ new seeds at $\mathcal{I}$.
2. For each seed
   (a) Project optic ray of seed on $\mathcal{I}$.
   (b) Detect largest local maximum x_{n+1} of NCC score within search window (section 4.2).
   (c) Update posterior parameters with new depth measurement x_{n+1} (section 3.2).
3. Remove all seeds with inlier ratio less than $\eta_{outlier}$.
4. Convert into 3d points (and remove from seed list) all seeds with inlier ratio higher than η_{inlier} and $\sigma_n < \epsilon$.

4.3. Pruning of seeds

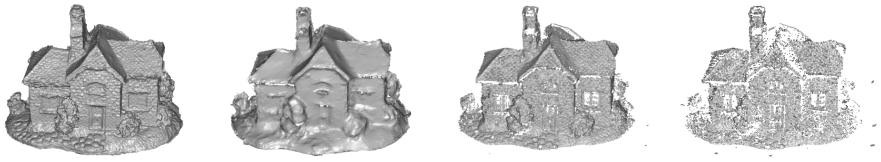
After the seed evolution step described in the previous section there are three possible outcomes:

- The seed has converged to a good estimate and therefore it is removed from the seed list and a 3d point is generated at the current posterior mean μ_n.
- The seed has failed to converge due to too many outliers present. The seed is then removed from the list.
- The seed has not been left to converge long enough and therefore it survives into the next evolution step.

To decide on the appropriate outcome we use the variance of the depth posterior σ_n^2 and the estimated inlier probability π. We employ the following criteria:

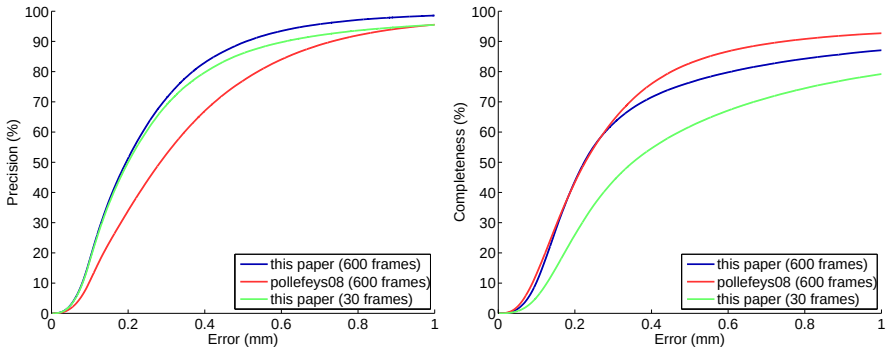
1. If according to our current posterior distribution $q\left(Z, \pi | a_n, b_n, \mu_n, \sigma_n^2\right)$ the inlier ratio π is less than $\eta_{outlier}$ with a probability of 99% then we can conclude that the depth estimation has failed. This is typically the case when the seed is initialized on an image patch that was out of focus, or there was not enough texture to match in subsequent images (Fig 5 b,c).
2. If the mean inlier ratio of our posterior is more than η_{inlier} and the depth variance σ_n is less than ϵ then we assume that the depth estimation has succeeded (Fig 5 a).
3. In all other cases we let the seed evolve further.

Throughout all our experiments the threshold parameters were kept fixed at $\eta_{outlier} = 0.05$, $\eta_{inlier} = 0.1$. The variance threshold ϵ was set at 1/10000th of the bounding volume size $Z_{max} - Z_{min}$. The generated 3d points are collected into an octree structure that is graphically rendered with z-buffer shading in real-time. Alg. 1 provides a summary of our method.



(a) (b) (c) (d)

Figure 6: **Comparison against ground-truth.** Our algorithm was compared with [22] on a 600 frame video sequence of a toy house. (a) Ground truth model of the house. (b) The result of [22].(c) Our result on the entire 600 frames. (d) Result of our method running on every 20 frames (total of 30 input images). Our results appear more detailed compared to [22] but especially in the case of the 30 frame result, less complete. This is due to the lack of any spatial regularization in our method as well as the fact that [22] produces a mesh while our results are 3d point-clouds. Full completeness-precision curves for these results can be found in figure 7.



(a) (b)

Figure 7: **Accuracy and completeness curves for ground truth experiment.** (a) shows accuracy results for our algorithm running in 30 and 600 frames of a video sequence of a house. The graph shows for a given distance d, how much of the reconstructed model falls within d of the ground truth. (b) measures completeness. I.e. for a distance d how much of the ground truth falls within d of the reconstructed model. Our results are more accurate but somewhat less complete. This is because our method performs no regularization and returns an unmeshed point-cloud.

completeness	50%	80%	85%	90%	95%	100%
Histograms (mm)	0.36	0.80	0.99	1.34	2.10	4.00
$Gauss + Uniform$ (mm)	0.33	0.70	0.84	1.05	1.57	3.42

Table 1: Accuracy/completeness comparison between histogram voting (first row) and our probabilistic sensor model on the Middlebury ground truth data [23] (second row). The probabilistic model is outperforming the histogram approach across all completeness levels.

5. Evaluation

Here we present the results of two evaluations of our method against ground truth data. In the first experiment we compare the histogram voting approach of [13] with our probabilistic formulation. We focus on depth estimation performance, isolating effects such as surface regularization or meshing. To that end, we generated depth estimates for 1.5 million pixels randomly selected from the 312 images of the 'fullTemple' sequence in the Middlebury evaluation [23]. For each pixel, we estimated its depth using our probabilistic formulation as well as the histogram voting approach. We then ran the standard completeness/precision tests on the two point-clouds. The results are summarized in table 1. The probabilistic formulation is outperforming the histogram approach across all completeness levels. This confirms that our model provides better depth estimation for the same data whilst offering the benefits of a probabilistic approach.

The second experiment involves comparing against [22], which is one of the few MVS methods that offer real-time performance. The subject is a small toy house which we reconstructed to a very high accuracy using a sequence of 36, 8-megapixel images and the publicly available PMVS [8] software. This reconstruction is treated as ground truth for the purpose of this experiment. We then captured a 600 frame video sequence of the same object and ran our method on the full video sequence as well a sub-sampled sequence of only 30 frames (skipping 19 out of every 20 frames) to evaluate how our method degrades with less data. We also asked the authors of [22] to run their algorithm on the same video sequence. The results are shown in figures 6 and 7. In summary, Compared to [22] our results are more precise (89.6% of our reconstruction falls within 5mm of the ground truth compared to 77.0% for [22]). However, because of our lack of regularization and the fact that [22] is providing a 3d surface our results suffer in completeness (We cover 76.4% of the ground truth surface compared to 82.7% for [22] for a distance threshold of 0.5mm). From the experiment on the subsampled sequence we note that our method degrades gracefully with less data. The reconstructed points we return are still accurate, however the algorithm manages to convert less seeds into 3d points, which leads to lower completeness figures.

Finally, figure 8 shows several challenging objects that were reconstructed by a user operating our system. The time required (including user time and computation time) was between 1 to 2 minutes per model. In the supplementary material we provide additional videos of our system in action.

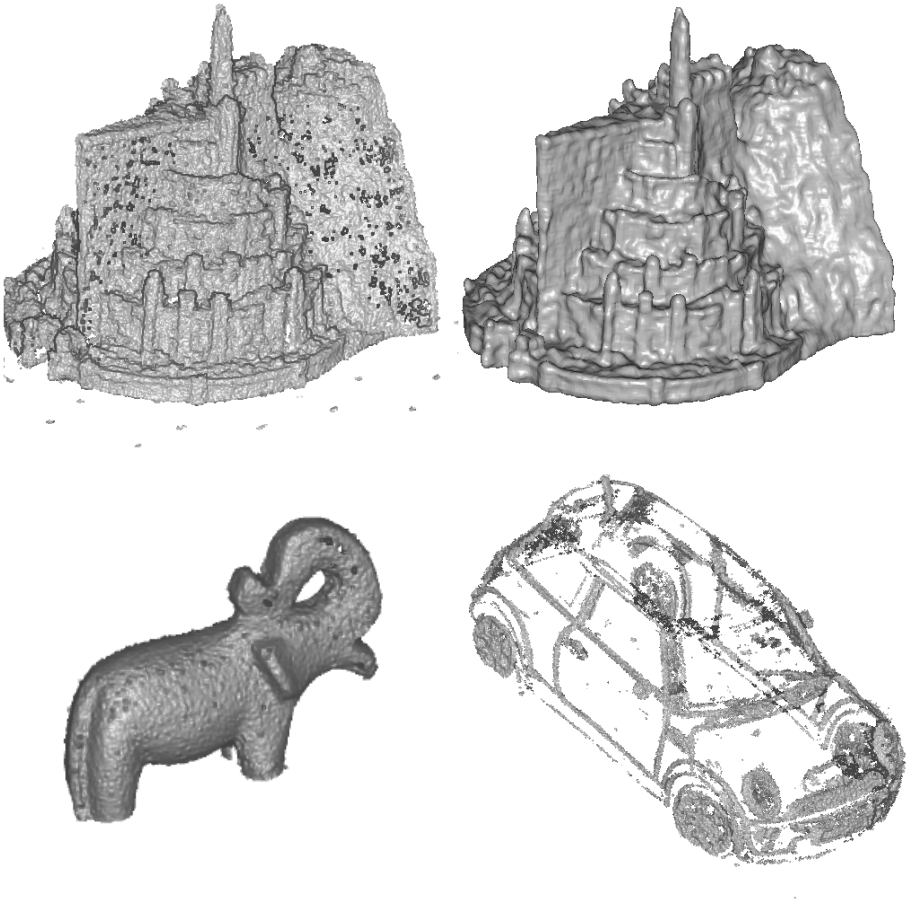


Figure 8: **Various 3d models acquired with our system.**
Top row: Left: raw point cloud. Right: a 3d mesh extracted using a graph-cut method similar to [15]. Bottom row: Left a well textured elephant figurine. Right: a textureless toy car. The model is incomplete but contains few spurious points.

6. Conclusion

This paper presented a video MVS method based on independent per-pixel depth estimation. We look for local maxima of correlation score along the epipolar line and fuse these candidate 3d locations within a probabilistic framework. Our implementation of this method can process 2-megapixel video at 60Hz, producing accurate reconstructions of small objects while providing an online feedback to the user. The validity of our approach was evaluated against ground-truth and was found to produce accurate reconstructions, degrading gracefully as the quantity of input data decreases.

One of the aims of this paper was to demonstrate the usability of video-based real-time MVS systems that provide an online feedback through intermediate results. To that end we showed how our method can be used to obtain 3d models of a variety of objects within a few seconds. Another aim was to evaluate how well does video data resolve ambiguities in MVS without any type of regularization. Our results show improvement in terms of accuracy compared to regularized methods, in exchange for lower completeness.

We believe that video-based MVS systems have great potential for reconstructing large-scale models when acquisition time is at a premium. This is because they provide a denser coverage of the object than still photographs while the online feedback helps avoid costly return visits to the scene. In future work we intend to verify this by deploying our method outdoors and applying it to the reconstruction of large scale scenes.

References

[1] S. Agarwal, N. Snavely, I. Simon, S. M. Seitz, and R. Szeliski. Building rome in a day. In *Proc. 12^{th} Intl. Conf. on Computer Vision (ICCV)*, 2009.

[2] R. Bhotika, D. Fleet, and K. Kutulakos. A probabilistic theory of occupancy and emptiness. In *Proc. 7^{th} Europ. Conf. on Computer Vision (ECCV)*, pages 112–130, 2002.

[3] D. Bradley, T. Boubekeur, and W. Heidrich. Accurate multi-view reconstruction using robust binocular stereo and surface meshing. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2008.

[4] A. Broadhurst, T. Drummond, and R. Cipolla. A probabilistic framework for space carving. In *Proc. 8^{th} Intl. Conf. on Computer Vision (ICCV)*, volume 1, pages 338–393, 2001.

[5] N. Campbell, G. Vogiatzis, C. Hernández, and R. Cipolla. Using multiple hypotheses to improve depth-maps for multi-view stereo. In *Proc. 10^{th} Europ. Conf. on Computer Vision (ECCV)*, 2008.

[6] N. Cornelis, B. Leibe, K. Cornelis, and L. Gool. 3d urban scene modeling integrating recognition and reconstruction. *Intl. Journal of Computer Vision*, 2-3(78):121–141, July 2008.

[7] A. Davison. Real-time simulataneous localisation and mapping with single camera. In *Proc. 9^{th} Intl. Conf. on Computer Vision (ICCV)*, 2003.

[8] Y. Furukawa and J. Ponce. Accurate, dense, and robust multi-view stereopsis. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2007.

[9] P. Gargallo and P. Sturm. Bayesian 3d modeling from images using multiple depth maps. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, volume 2, pages 885–891, jun 2005.

[10] M. Goesele, B. Curless, and S. Seitz. Multi-view stereo revisited. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, volume 2, pages 2402–2409, 2006.

[11] M. Goesele, N. Snavely, B. Curless, H. Hoppe, and S. M. Seitz. Multi-view stereo for community photo collections. In *Proceedings of ICCV 2007*, October 2007.

[12] M. Habbecke and L. Kobbelt. A surface-growing approach to multi-view stereo reconstruction. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2007.

[13] C. Hernández and F. Schmitt. Silhouette and stereo fusion for 3d object modeling. *Computer Vision and Image Understanding*, 96(3):367–392, December 2004.

[14] C. Hernández, G. Vogiatzis, and R. Cipolla. Probabilistic visibility for multi-view stereo. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2007.
[15] A. Hornung and L. Kobbelt. Robust reconstruction of watertight 3d models from non-uniformly sampled point clouds without normal information. In *SGP '06*, pages 41–50, 2006.
[16] E. Jaynes. *Probability Theory, The Logic of Science*. Cambridge University Press, 2003.
[17] G. Klein and D. Murray. Improving the agility of keyframe-based slam. In *Proc. 10^{th} Europ. Conf. on Computer Vision (ECCV)*, 2008.
[18] K. Kolev and D. Cremers. Integration of multiview stereo and silhouettes via convex functionals on convex domains. In *Proc. 10^{th} Europ. Conf. on Computer Vision (ECCV)*, 2008.
[19] K. N. Kutulakos and S. M. Seitz. A theory of shape by space carving. *Intl. Journal of Computer Vision*, 38(3):199–218, 2000.
[20] M. Lhuillier and L. Quan. A quasi-dense approach to surface reconstruction from uncalibrated images. *IEEE Trans. Pattern Anal. Mach. Intell.*, 27(3):418–433, 2005.
[21] X. Li, C. Wu, C. Zach, S. Lazebnik, and J.-M. Frahm. Modeling and recognition of landmark image collections using iconic scene graphs. In *ECCV '08: Proceedings of the 10th European Conference on Computer Vision*, pages 427–440, 2008.
[22] M. Pollefeys et al. Detailed real-time urban 3d reconstruction from video. *Intl. Journal of Computer Vision*, 78(2-3):143–167, 2008.
[23] S. Seitz, B. Curless, J. Diebel, D. Scharstein, and R. Szeliski. A comparison and evaluation of multi-view stereo reconstruction algorithms. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages 519–528, 2006.
[24] N. Snavely, S. M. Seitz, and R. Szeliski. Photo tourism: Exploring image collections in 3d. *ACM Transactions on Graphics (Proceedings of SIGGRAPH 2006)*, 2006.
[25] C. Strecha, R. Fransens, and L. V. Gool. Wide-baseline stereo from multiple views: A probabilistic account. *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, 1:552–559, 2004.
[26] C. Strecha, W. von Hansen, L. V. Gool, P. Fua, and U. Thoennessen. On benchmarking camera calibration and multi-view stereo for high resolution imagery. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2008.
[27] G. Vogiatzis, C. Hernández, P. Torr, and R. Cipolla. Multi-view stereo via volumetric graph-cuts and occlusion robust photo-consistency. *IEEE Trans. Pattern Anal. Mach. Intell.*, 29(12):2241–2246, December 2007.
[28] Z. Zhang. A flexible new technique for camera calibration. *IEEE Trans. Pattern Anal. Mach. Intell.*, 22(11):1330–1334, 2000.