

Complementary geometric and optical information for match-propagation-based 3D reconstruction

Patricio Alejandro Galindo, Rhaleb Zayer

▶ **To cite this version:**

Patricio Alejandro Galindo, Rhaleb Zayer. Complementary geometric and optical information for match-propagation-based 3D reconstruction. Proceedings on ACCV 2014, Nov 2014, Singapore, Singapore. hal-01088426

HAL Id: hal-01088426
https://inria.hal.science/hal-01088426

Submitted on 27 Nov 2014

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Complementary geometric and optical information for match-propagation-based 3D reconstruction

Patricio A. Galindo and Rhaleb Zayer

INRIA Nancy, France

Abstract. In this work, we consider the problem of propagation-based matching for 3D reconstruction, which deals with expanding a limited set of correspondences towards a quasi-dense map across two views. In general, propagation based methods capture well the scene structure. However, the recovered geometry often presents an overall choppy nature which can be attributed to matching errors and abrupt variations in the estimated local affine transformations. We propose to control the reconstructed geometry by means of a local patch fitting which corrects both the matching locations and affine transformations throughout the propagation process. In this way, matchings that propagate from geometrically consolidated locations bring coherence to both positions and affine transformations. Results of our approach are not only more visually appealing but also more accurate and complete as substantiated by results on standard benchmarks.

1 Introduction

This paper revisits the problem of match propagation across a pair of views in the context of quasi-dense matching [1], and its extension to wide-baseline views which accounts for affine distortion [2]. We focus on these approaches due to their inherent simplicity and minimal requirements, however the proposed solutions can be applied to other propagation approaches. Alternative approaches such as [3, 4] which require image rectification and further problem reformulations will not be addressed here.

Propagation-based matching is the workhorse of many surface reconstruction approaches e.g. [5–8]. In practice, it comes in different flavors, e.g. [1, 9, 10], but they all require corrective steps and/or postprocessing techniques to filter out mismatches. Furthermore, the resulting geometry exhibits a choppy nature which is often not addressed directly and is left to mesh reconstruction algorithms, e.g. [11]. Figures 1 and 2 show typical chaps and clefts observed when performing scene reconstruction using a state of the art propagation approach, in this case [2]. We are aware that the use of additional images can help reduce such effects relatively, e.g. [12, 7], however, addressing the challenges of the fundamental problem on a pair of images remains necessary to take full advantage of the available data and can also be beneficial when additional views are accessible.

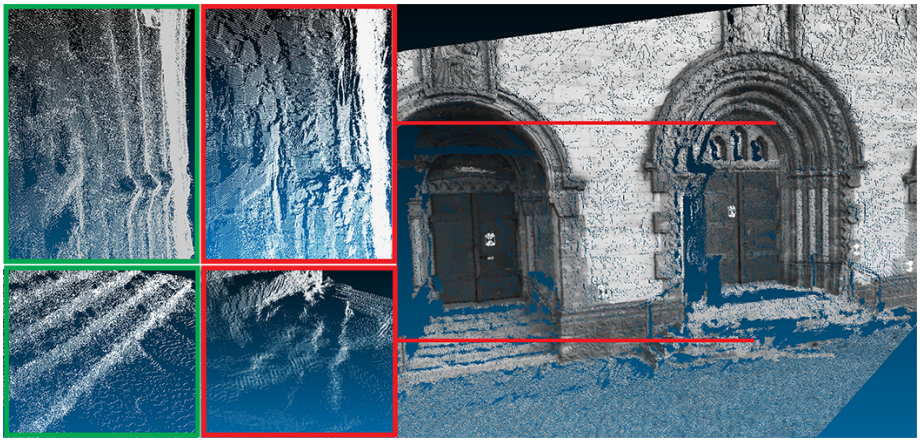


Fig. 1: Reconstruction from views 3-4 of the Herz-Jesu P8 dataset using [2] (right). Geometric artifacts are clearly visible in the closeups (middle). Whereas, the geometry of the scene is better captured with our approach (left).

In this paper, we argue that such problems can be resolved during the propagation process by including geometric cues which help consolidate the geometric reconstruction and the estimation of the local affine transformations, thus avoiding tedious post processing operations in the first place. The use of geometric information in propagation based methods is not new, for instance [13], proposes a splatting inspired approach based on local plane fitting. However, this approach is limited to narrow baselines and requires a relatively large number of images. Our approach, on the other hand, proceeds by fitting small surface patches on which initial 3D points are projected to adjust the matches and the local affine transformations. By operating in such way, the propagated information is confirmed by both optical and geometrical cues throughout the whole matching process.

In section 2, we outline the propagation algorithm using only optical information and in section 3, we proceed to outline and detail our approach. In section 4, results are shown and discussed.

2 Background

To keep this exposition succinct and self contained, we briefly summarize the key ideas in quasi-dense propagation and we adopt the formalisms of [1, 2]. The reader is referred to those papers and the references therein for a detailed description. Match propagation aims to obtain a larger number of matches between two views (I, I') starting from an initial set of matches (seeds) $S = \{\mathbf{s}(\mathbf{x}, \mathbf{x}'), \mathbf{x} \in \mathbf{I},\ \mathbf{x}' \in \mathbf{I}'\}$. When dealing with wide-baseline views, it is preferable to have the fundamental matrix between the views as well as the local affine transforms A associated with the individual seeds [2]. The affine transforms can



Fig. 2: Reconstruction from views 2-3 of the Fountain P11 dataset using [2] (left). Geometric artifacts are clearly visible in the closeups (middle). Whereas, the geometry of the scene is better captured with our approach (right).

be obtained using existing affine covariant region detectors [14]. In order to reduce geometric distortions, the reference view for each seed is defined so as the affine transformation across the image pair is always magnifying. Similarity is measured using the zero-mean normalized cross-correlation (ZNCC) of geometrically normalized image patches. Once the initialization data is set, the match propagation operates on a priority queue based on correlation scores of the seeds and proceeds by repeating the following steps:

- remove the seed s with best score from queue
- search new candidates in a neighborhood of s based on local affine transform
- compute correlation scores for all candidates
- append candidates to the matches and seeds if they have high correlation score, satisfy the epipolar constraint and are not yet filled
- update candidate's affine transform based on optical information
- mark the corresponding pixels as filled

3 Geometry based image match propagation

3.1 Overview

The central idea of our approach is to couple geometric and optical information in a complementary fashion, in the sense that they correct and confirm each other. In this regard, when a sufficient number of data points are available in a localized region, a surface fitting can be performed and subsequently the matches and their affine transformation will be updated geometrically.

In practice, three main recurrent stages take place in our method: propagation, region search, and update by surface fitting. In the next paragraphs we briefly describe these stages and we later explain them in greater algorithmic detail in the next sub-sections.

i) Propagation: In a propagation step the best seed (s) from the queue is selected and new candidates within a neighborhood of s are obtained using its corresponding affine transform. Candidates with correlation scores which pass a minimum threshold and comply with map occupancy and epipolar constraints are added to the queue (see figure 3) and are marked as unconfirmed matches.

ii) Best candidate region querying: When new matches are found in the propagation step, the search for a suitable region for local surface fitting is initiated. This search is performed by sliding corresponding windows in the two image planes (depicted by the windows in figure 5). Candidate regions where the data points provide enough support for surface fitting are validated and sorted giving preference to the ones with the most confirmed points.

iii) Local fitting and re-projection: When a suitable region is found in the previous step a local surface patch fitting is performed. A sub-set of the points representing the core of the region (see figure 5), are then projected onto the local surface patch. These new positions are re-projected to the image planes to update the correspondences.

Points that re-project too far in the image planes or yield a large change in their affine transformation remain marked as unconfirmed. Matches which improve their correlation scores are updated and marked as confirmed. This update serves the purpose of correcting point locations as well as steering subsequent propagation (see figure 4).

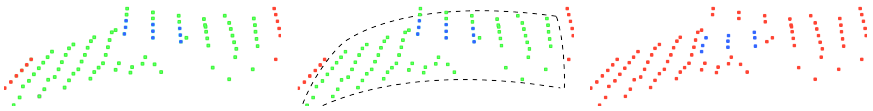


(a) Propagation step n (b) Propagation step n+1 (c) Propagation step n+2

Fig. 3: Illustration of a few propagation steps before performing any geometric updates.

3.2 Propagation

In principle, the propagation stage proceeds similarly to the wide baseline matching proposed in [2] but presents three important differences. **First**, newly propagated matches are not defined as final matches; instead, they are stored as



(a) Candidate region selected (b) Local surface fitting (c) New propagation replaces initial points

Fig. 4: Illustration of the geometric fitting steps. First, the candidate region is selected (a), the support and the core are colored in green and blue respectively. Second, local surface fitting is performed (b). Third, the core points are projected onto the surface (c).

unconfirmed matches. **Second**, the seeds sorting operation gives preference to geometrically confirmed seeds and checks for correlation score in a secondary sorting step. **Third**, new matches stemming from a geometrically confirmed seed can replace previous unconfirmed matches if their correlation score is higher. Note that this second set of conditions do not come into play from the beginning since it takes several propagation steps to obtain a set of geometrically confirmed matches. This categorization of points is depicted in figure 5 where blue indicates confirmed points and red indicates unconfirmed (replaceable) ones. In order to fix the ideas, we describe the steps of our propagation stage in algorithms 1 and 2. Note that the 3D positions originating from each pair of matches are obtained at this stage since they are needed for the later region search and surface fitting.

3.3 Candidate region querying

The goal at this stage is to identify regions with adequate point distribution and density to support a local patch fitting. The region querying is initiated once a set of new unconfirmed points are obtained. In order to steer clear from problematic regions during this search we avoid: i) areas which present large jumps, ii) points which do not remain within their respective patch perimeter when projected on the corresponding images planes. These requirements help prevent fitting and consequentially smoothing sharp features, e.g. the stairs in figure 9. On the other hand, we favor regions that present the most confirmed points so that the large bodies of confirmed points expand first instead of creating several isolated clusters of confirmed points. A typical scenario is shown in figure 5 where a set of confirmed matches (in blue) is surrounded by a set of unconfirmed points (in red). The green windows represent the support region (outer square) and the core window (inner square). Only unconfirmed points inside the core window will be fit in the next stage. In the same figure, the points inside the orange region represent newly matched points that enable a surface to be fit at the depicted location.

The conditions that define wether surface fitting can be performed are:

Input: Set of seeds with local affine transforms, projection matrices P_1,P_2
Output: Matches, Seeds, 3D point Cloud

while *Seeds* $\neq \emptyset$ **do**
  Sort(Seeds);
  foreach *seed* (x, x') **do**
    Local $= \emptyset$;
    `/* Local stores new candidates */`
    foreach (u, u') *in* $N(x, x')$ **do**
      if $SampsonDistance(u, u') < sd$ *and ((*$map_1(u)$ *and* $map_2(u')$ *are not filled) or (s is confirmed and correspondence occupying* $map_1(u)$ *and* $map_2(u')$ *is not))* **then**
        Compute $z = ZNCC(u, u')$ and standard deviations d, d';
        if $z > min_correlation$ *and* $d > t$ *and* $d' > t$ **then**
          store (u,u',z) in Local;
        end
      end
    end
    sort(Local);
    foreach $L(u, u')$ *in Local* **do**
      if $map_1(u)$ *and* $map_2(u')$ *are not taken* **then**
        $X \leftarrow Compute3D(u_1, u_2, P_1, P_2)$;
        $A_O \leftarrow$ UpdateAffineEstimationOptical(u,u');
        mark $map_1(u)$ and $map_2(u')$ as taken by an unconfirmed point;
        store (u,u',A_O) in Seeds;
        store (u,u',A_O) in Matches;
        store (X) in Cloud;
      end
    end
    while $R \leftarrow$ *BestCandidateRegionQuerying(*map_1,map_2,$Cloud$,$Matches$*) exists* **do**
      FitSurfacePatch(R,$Cloud$,$Seeds$,$Matches$);
    end
  end
end

Algorithm 1: Main match propagation. Parameter values used in all our experiments: $min_correlation = 0.75$, $t = 2$, $sd = 1$. The FitSurfacePatch is summarized in algorithm 2. The SampsonDistance is defined as in [15] and the Sort function follows the conditions set in 3.1

- points inside the support and core regions defined on the first image should also lay inside the corresponding support and core regions defined on the second image
- there should be at least one point in the core that was not yet confirmed
- regions inside the support window but to the north, south, east and west of the core window should be sufficiently populated (at least 50%)
- points inside the core window should be dense enough, any $2x2$ window should contain at least one match(pixels)

The sizes of the windows in all of our experiments are 15×15 for the support and 5×5 for the core. These windows are defined in the view that presents the most fronto-parallel surface with respect to the image plane and are transformed to the other view using the affine transformation estimation of the point at the center of the window.

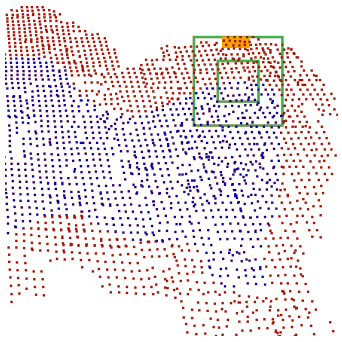


Fig. 5: Searching for best region to fit a surface patch. Confirmed points are depicted in blue, unconfirmed ones in red. Points in the orange region were obtained in the preceding propagation step. In green, support (outer) and core (inner) windows.

3.4 Fit surface and update

In this stage, we perform a surface fitting to the selected points inside a region obtained in the previous stage. For this purpose we use a least squares quadratic polynomial fitting of the form $ax^2 + by^2 + cxy + dx + ey + f$ [16]. Although it is possible to perform higher order fitting, we found in our experiments that quadratic fitting is fairly satisfactory for our purpose. The selected points used in this fitting stage are either part of the core region or of the support region. The core region encompasses a small set of points that are surrounded by a set of support points; this is because points should be fit and projected only when there is enough information to describe the local surface.

During the fitting process, the impact of each point in the least squares formulation can be controlled through a set of weights [16]. We formulate the weights as the product of three values: the correlation score of the corresponding matching points, the inverse 3D distance of the reconstructed point to the center of the region, and the nature of the point (confirmed or unconfirmed). More specifically, $w = z^3 \times (1 + is_confirmed)/distance(X, X_c)$, where z is elevated to the third power in order to stretch the range. Once the surface is fit, each 3d point p is projected to a point p' that lies on the fitted surface.

Each newly obtained point p' can then be reprojected to the image planes I and I', obtaining the respective matching image positions u_r and u'_r. Furthermore, an estimation of the associated affine transformation is recovered. This

affine transformation estimation (A_G) is obtained by sampling two additional points around p', projecting them to the image planes and computing the affine transformation that describes their local motion using the re-projection of p' as center of coordinates. Once these new estimations are obtained, we first require that the new 2D matching points induce only a small adjustment of the initial matching positions. Namely, $distance(u_r, u) < \gamma$ and $distance(u, u'_r) < \gamma$ (with $\gamma = 1.5$ pixels used in all our experiments). In this manner we ensure that points near surface edges do not drift away from their original location. Next, we verify if A_G does not represent a large change in transformation estimation with respect to the current estimation A. At the same time we need to verify that A_G does not yield a large shear. Specifically, we verify if $\theta \leq det(A)/det(A_G) \leq 1/\theta$, with $\theta = 0.5$ for all of our cases. We then define the eigen-values rations $eigratio_g = \frac{eig_1(A_G)}{eig_2(A_G)}$ and $eigratio = \frac{eig_1(A)}{eig_2(A)}$ and verify if $\theta \leq eigratio_g \leq 1/\theta$ and if $\theta \leq eigratio_g/eigratio1/ \leq \theta$. A point will remain as unconfirmed is it does not fulfill all of these requirements. Note that the common use of θ for all the previous tests is not required. Independent values can be used for the three verifications but in all of our experiments the same value was used.

The steps described above are summarized in algorithm 2.

Input: region R, list of Matches, point Cloud, list of Seeds
Output: Updated: Cloud, Matches and Seeds
$w \leftarrow (z^3 \times 1/distance(X, X_c) \times (1 + is_confirmed))$;
perform least squares surface fitting to obtain surface Q;
foreach *3D point p inside the core of region R* **do**
  obtain p' by projecting p to surface Q;
  if $distance(p', p) < \epsilon$ **then**
    $p'_1 \leftarrow$ re-project p' to I;
    $p'_2 \leftarrow$ re-project p' to I';
    $A_G \leftarrow$ compute affine transformation using Q and p';
    if *A_G does not represent a large change from A and does not represent too much distortion* **then**
      Obtain normalized windows using A_G;
      $z_new \leftarrow ZNCC(p'_1, p'_2)$;
      if $z_new > z_old$ **then**
        $UpdateCloud(p')$;
        $UpdateMatches(p'_1, p'_2, A_G)$;
        $UpdateSeeds(p'_1, p'_2, A_G)$;
        mark $map_1(p'_1)$ and $map_2(p'_2)$ as taken by a confirmed point;
      end
    end
  end
end

Algorithm 2: *FitSurfacePatch* algorithm that performs matching correction by surface fitting and re-projection

4 Results

We tested our algorithm on various data sets comprising standard benchmarks as well as in-house acquired data. Typical results of our approach compared to those of Kannala and Brandt [2], Brox and Malik [17] and Tola et al. [18] are shown for the views 2-3 of the Fountain-P11 dataset (figures 6 and 7), and for the views 3-4 of the Herz-Jesu-P8 dataset (figures 8 and 9). Both datasets are provided in [19]. We benchmarked our results measuring depth estimation error with respect to the ground truth, in a similar way as in [12]. The benchmarked results are presented in figures 6 and 8 as error vs occupancy histogram graphs and as color coded depth estimation errors. We use the code provided by the authors of [2] to perform comparisons and we used similar values for the common parameters of the algorithms. We also use the code provided by the author of the variational matching method [17]. For the case of [18], we use the code provided by the author to compute the DAISY dense descriptor in combination with our own implementation of a descriptor matching method that searches for matches within rectified images. Since such approach does not accurately represents what is proposed in [18], we also compare our results to those presented in the later paper, following their own evaluation metrics (see figure 12). Notice that in both quantitative evaluations our method achieves better results.

In all cases our approach performs better than the state of the art pairwise propagation approach [2] and the other methods here tested. The depth error benchmarks presented in figures 6 and 8 clearly show that our method presents more accurate and more complete results.

In figure (6-top-right) it can be observed that our method attains 74.2% of the viewed Fountain scene with an error of less than 0.1; while the method of Kannala and Brandt achieves only 68.3%. In the same error vs occupancy graph we oppose our results to the combination of [2] with an MLS [16] in post-processing. This combination leads only to a small improvement for errors in the lower range but an overall decay in the quality of the results is observed.

In figure (8-top-right) our method's score for errors lower than 0.1 in the viewed Herz-Jesu scene is 63.6%. The method of [2] as well as its combination with MLS post-processing yield only 58.4% and 50.9% of occupancy respectively.

The closeups to the fountain result presented in figure 7 illustrate the improvements gained by using our method. Our results do not exhibit the choppy characteristic of previous work and present a clearer and more accurate structure of the scene. The closeups to the Herz-Jesu result presented in figure 9 not only show that the quality is improved but also the total coverage. In this example the main stairs in the scene are accurately reconstructed thanks to the continuous interpretation and improvement of the matching results that lead to better propagations.

We also performed a comparison using two views of our own face dataset that comprises 1.3 mega-pixels images and two views (7 and 15) of the warrior dataset from [20]. Results are presented in figures 10 and 11 were uncolored closeups are presented next to the full views of the results. In both cases the results are significantly better when our approach is used.

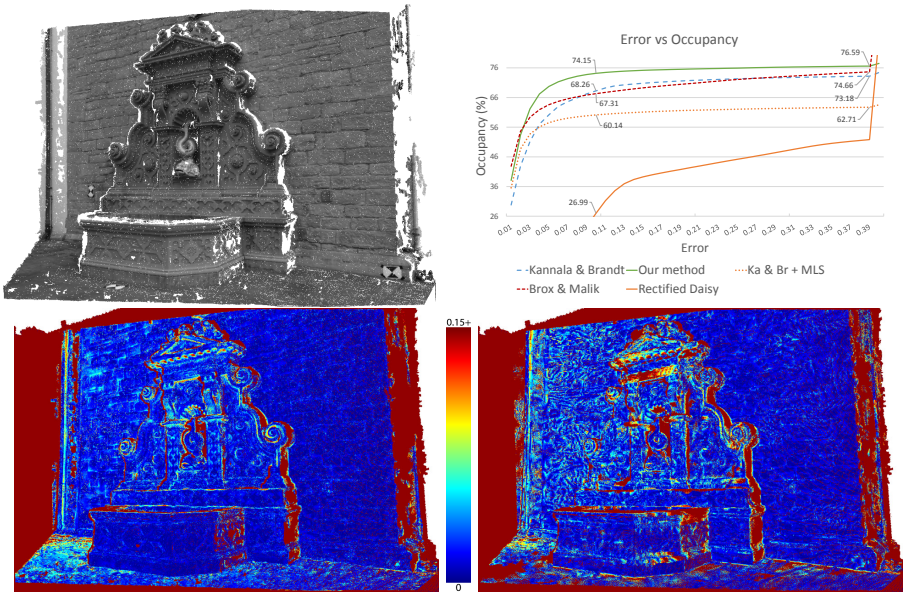


Fig. 6: Results of our approach on the Fountain P11 views 2-3 (top-left). On the top-right, a chart illustrating the depth error vs occupancy of our algorithm, [2], [17], [18] (rectified) and [2] with MLS as post-processing.On the bottom color coded depth error using our algorithm (left) and using [2] (right).

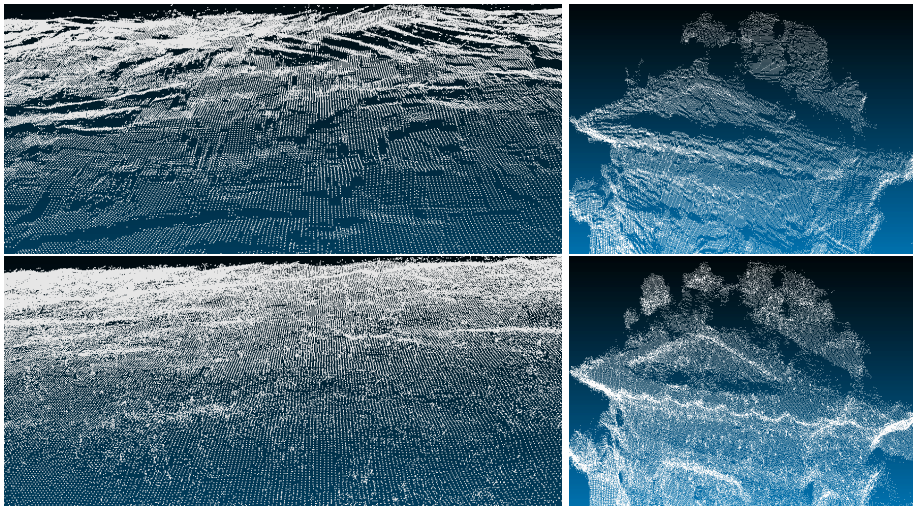


Fig. 7: Closeups on portions of the wall and fountain-top for the Fountain P11 views 2-3 using [2] (top) and using our method (bottom)

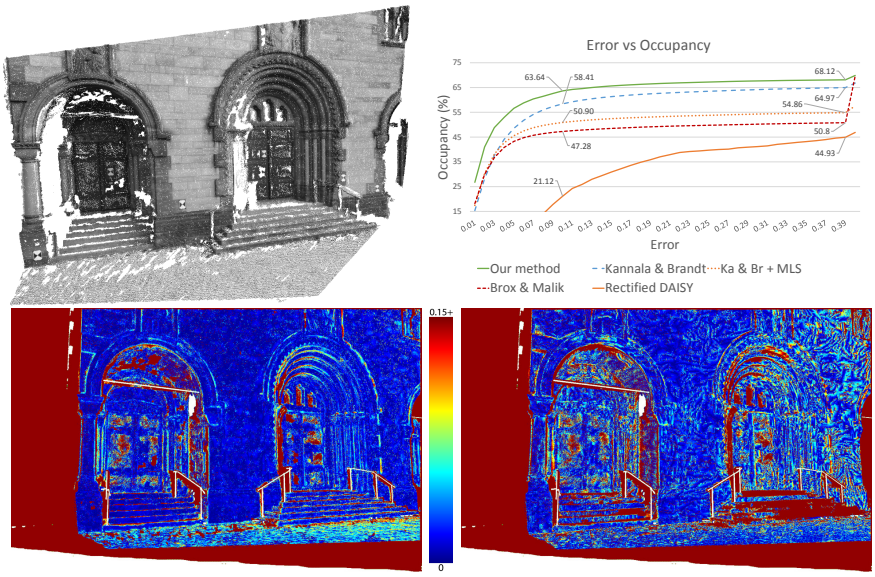


Fig. 8: Result of our approach on the Herz-Jesu P8 views 3-4 (top-left). On the top-right, a chart illustrating the depth error vs occupancy of our algorithm, [2], [17], [18] (rectified) and [2] with MLS as post-processing.On the bottom, color coded depth error using our algorithm (left) and using [2] (right).

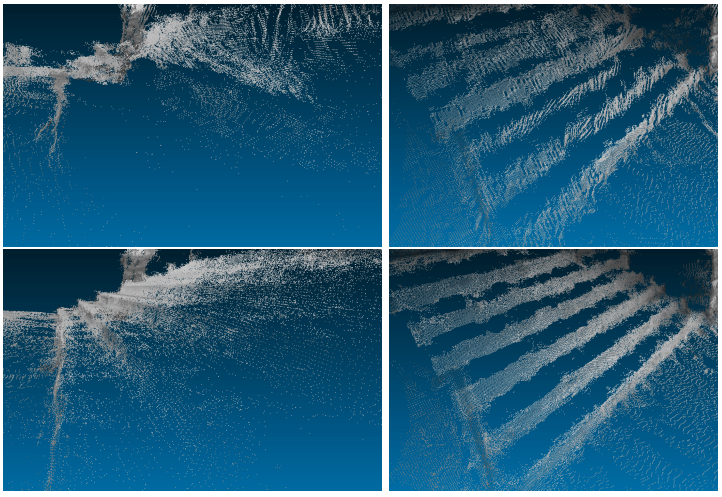


Fig. 9: Closeups on to results on Herz-Jesu P8 views 3-4 using [2] (top) and using our method (bottom). A closeup to a portion of the main stairs (left) and to the left-most stairs in the scene (right)



Fig. 10: Results on two views of our face dataset. The two views used (top), our result with a closeup to the cheek (middle), results using [2] (bottom)

5 Conclusion and discussion

This paper presented propagation approach which congruently uses optical and geometric information to steer the propagation process. Although, many of underlying ideas are fairly simple, they lead to significant improvements both in the accuracy and completeness. Furthermore, as it could be argued that performing an equivalent MLS fitting as post processing would yield equivalent results, our

comparisons to such a scenario, suggest that actually the results get worse. The reason being that in our case the subsequent matches depend on the fitting, whereas the use of fitting in post processing cannot alter how the matches are propagated.

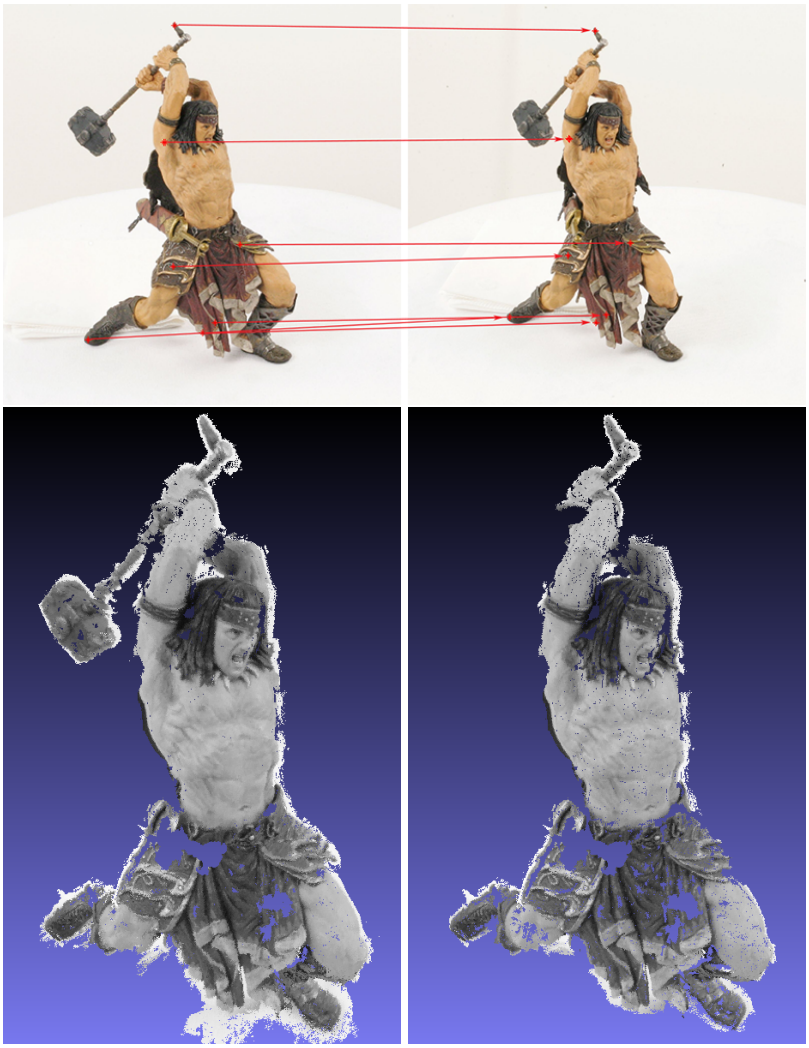


Fig. 11: Results on views 7 and 15 of the warrior dataset from [20]. The two views used are showed at the top row along with the 7 features matched. On the bottom row, our result (left) and results using [2] (right). Notice that our method returns less holes and it is able to reconstruct the warrior's hammer.

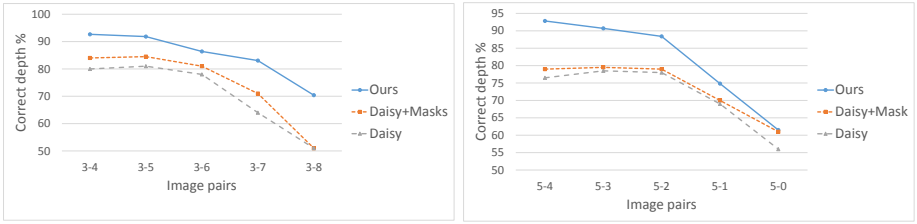


Fig. 12: Comparisons of our results to those of DAISY [18]. Results on the Fountain scene (left) and the Herz-Jesu scene (right) for pairs of images with increasing baselines. The charts present the percentage of correct depth estimations for each pair of views. A depth estimation is considered correct if it presents and error of less than 1% of the scene's depth range [18] when compared to the laser scanned data.

Acknowledgments The authors would like to thank, the anonymous reviewers for their feedback on the paper, Juho Kannala for making his code publicly available, Christoph Strecha and Yasutaka Furukawa for their Multi-View datasets and Samuel Hornus for the face dataset. This work was funded by the ANR (Agence Nationale de la Recherche) under grant (PhysiGrafix ANR-09-CEXC-014-01).

References

1. Lhuillier, M., Quan, L.: Quasi-dense reconstruction from image sequence. In: ECCV. ECCV '02, London, UK, UK, Springer-Verlag (2002) 125–139
2. Kannala, J., Brandt, S.: Quasi-dense wide baseline matching using match propagation. In: CVPR. (2007) 1 –8
3. Chen, Q., Medioni, G.: A volumetric stereo matching method: Application to image-based modeling. In: CVPR. Volume 1., Fort Collins, United States (1999) 1029–1034
4. Cech, J., Sara, R.: Efficient sampling of disparity space for fast and accurate matching. In: CVPR. (2007) 1 –8
5. Lhuillier, M., Quan, L.: A quasi-dense approach to surface reconstruction from uncalibrated images. PAMI **27** (2005) 418–433
6. Yao, J., Cham, W.K.: 3d modeling and rendering from multiple wide-baseline images by match propagation. Signal Processing: Image Communication **21** (2006) 506 – 518
7. Furukawa, Y., Ponce, J.: Accurate, dense, and robust multi-view stereopsis. In: CVPR. (2007) 1 –8
8. Koskenkorva, P., Kannala, J., Brandt, S.: Quasi-dense wide baseline matching for three views. In: ICPR. ICPR '10, Washington, DC, USA, IEEE Computer Society (2010) 806–809
9. Otto, G., Chau, T.: A 'region-growing' algorithm for matching of terrain images. Image and Vision Computing **7** (1989) 83 – 94

10. Ferrari, V., Tuytelaars, T., Gool, L.: Simultaneous object recognition and segmentation by image exploration. In Pajdla, T., Matas, J., eds.: ECCV. Volume 3021 of Lecture Notes in Computer Science. Springer Berlin Heidelberg (2004) 40–54
11. Kazhdan, M., Bolitho, M., Hoppe, H.: Poisson surface reconstruction. In: Proceedings of the Fourth Eurographics Symposium on Geometry Processing. SGP '06, Aire-la-Ville, Switzerland, Switzerland, Eurographics Association (2006) 61–70
12. Ylimaki, M., Kannala, J., Holappa, J., Heikkila, J., Brandt, S.: Robust and accurate multi-view reconstruction by prioritized matching. In: ICPR, IEEE (2012) 2673–2676
13. Habbecke, M., Kobbelt, L.: A surface-growing approach to multi-view stereo reconstruction. In: CVPR. (2007) 1–8
14. Mikolajczyk, K., Tuytelaars, T., Schmid, C., Zisserman, A., Matas, J., Schaffalitzky, F., Kadir, T., Gool, L.V.: A comparison of affine region detectors. IJCV **65** (2005) 43–72
15. Hartley, R.I., Zisserman, A.: Multiple View Geometry in Computer Vision. Second edn. Cambridge University Press (2004)
16. Barnhill, R., Farin, G., Jordan, M., Piper, B.: Surface/surface intersection. Computer Aided Geometric Design **4** (1987) 3–16
17. Brox, T., Malik, J.: Large displacement optical flow: Descriptor matching in variational motion estimation. PAMI **33** (2011) 500–513
18. Tola, E., Lepetit, V., Fua, P.: Daisy: An efficient dense descriptor applied to wide-baseline stereo. PAMI **32** (2010) 815–830
19. Strecha, C., Von Hansen, W., Van Gool, L., Fua, P., Thoennessen, U.: On benchmarking camera calibration and multi-view stereo for high resolution imagery. In: CVPR. (2008) 1–8
20. Furukawa, Y., Ponce, J.: 3d photography dataset. (http://www.cse.wustl.edu/ furukawa/research/mview/index.html) Accessed: 2014-05-05.