

Abstract

Title of dissertation: Dense Wide-Baseline Stereo with Varying Illumination and its Application to Face Recognition

Carlos D. Castillo,
Doctor of Philosophy, 2012

Dissertation directed by: Professor David W. Jacobs
Department of Computer Science

We study the problem of dense wide baseline stereo with varying illumination. We are motivated by the problem of face recognition across pose. Stereo matching allows us to compare face images based on physically valid, dense correspondences. We show that the stereo matching cost provides a very robust measure of the similarity of faces that is insensitive to pose variations. We build on the observation that most illumination insensitive local comparisons require the use of relatively large windows. The size of these windows is affected by foreshortening. If we do not account for this effect, we incur misalignments that are systematic and significant and are exacerbated by wide baseline conditions.

We present a general formulation of dense wide baseline stereo with varying illumination and provide two methods to solve them. The first method is based on dynamic programming (DP) and fully accounts for the effect of slant. The second method is based on graph cuts (GC) and fully accounts for the effect of both slant and tilt. The GC method finds a global solution using the unary function from the general formulation and a novel smoothness term that encodes surface orientation.

Our experiments show that DP dense wide baseline stereo achieves superior performance compared to existing methods in face recognition across pose. The experiments with the GC method show that accounting for both slant and tilt can improve performance in situations with wide baselines and lighting variation. Our formulation can be applied to other more sophisticated window based image comparison methods for stereo.

DENSE WIDE-BASELINE STEREO WITH VARYING
ILLUMINATION AND ITS APPLICATION TO
FACE RECOGNITION

by

Carlos D. Castillo

Dissertation submitted to the Faculty of the Graduate School of the
University of Maryland, College Park in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
2012

Advisory Committee:

Professor David W. Jacobs, Chair/Advisor

Professor Min Wu, Dean's Representative

Professor Rama Chellappa

Professor Larry S. Davis

Professor Hal Daumé

© Copyright by
Carlos D. Castillo
2012

Dedication

A mi padre, quién con su ejemplo, su apoyo y su amor me ha inspirado a luchar hasta cansancio para obtener las cosas que quiero. Gracias por todo.

Acknowledgements

First and foremost I would like to thank my advisor, Prof. David W. Jacobs for his superb guidance and advice. Day in and day out for six years David amazed me with his brilliance, inspired me with his enthusiasm and surprised me with his generosity and his kind heartedness.

I would like to thank the members of my committee: Prof. Chellappa, Prof. Davis, Prof. Daumé and Prof. Wu for their insights and valuable comments regarding this work.

In my time at Maryland I met great people. I would like to thank all my academic siblings and lab and office mates through the years. In particular, I would like to thank Joao Soares, Anne Jorstad, Daozheng Chen, Arijit Biswas and Leonardo Claudino. I now consider them my friends and most importantly if it weren't for them I would had lost hope in the social abilities of all vision grad students!

I would now like to thank my non-vision-related friends, while they all now live in different cities, I will proceed to thank them all in the same paragraph. I would like to thank Grecia Lapizco-Encinas and her husband Marko Tadjer for their support and all the help I received from them when I really needed it. I would like to thank Gerardo Simari and Vanina Martínez for their friendship and for putting up with me and always cheering me up. I would like to thank Tony Vidal and Mary Pombo for their support and for all the grown up guidance they've given me (that includes, but is not limited to tax advice and Costco cards). I would like to thank my friend Eduardo "Hawaii" Ruiz for putting up with me for the last decade or so. I would like to thank Alejandro Rodríguez for his friendship and for being an example of all the things I could accomplish in the next five years, and he's only

two years older than me. I would like to thank all my friends, also, for all the good times we had in the past and all the good times I'm sure we'll have in the future.

Prof. Carolina Chang and Prof. Maria-Esther Vidal were instrumental on my coming to Maryland. Without their advice, encouragement, and friendship I wouldn't have started my Ph.D. in the first place.

I would would like to thank my parents, Carlos Alberto Castillo and Fátima Katiuska Bueno, for all the years of hard work they put into raising me and for encouraging me to work hard to obtain whatever I want in life and supporting me unconditionally in the process. I would like to thank my brothers Julio Castillo and Simón Castillo for all their help and encouragement in this and many other projects. Many times it was their (my parents and my brother's) unwavering faith in me that kept me going.

Finally, I would like to thank my wife Gabriela for all her love and all her support and for always believing in me even at times when I didn't.

Contents

1	Introduction	1
1.1	Stereo for Face Recognition	2
1.2	Improving on Stereo for Face Recognition	5
1.3	Advancing Towards Dense Wide-baseline Stereo Under Varying Illumination	7
1.4	Organization	8
2	Related Work	10
2.1	Face Recognition Across Pose	10
2.2	Face Recognition Across Illumination	15
2.3	Unconstrained Face Recognition	16
2.4	Stereo Matching with Slant	19
2.5	MRF Stereo	21
3	Stereo for Face Recognition Across Pose	23
3.1	Analysis of Stereo Matching for Face Recognition	23
3.2	Alignment	30
3.2.1	Epipolar Geometry under Scaled Orthographic Projection	31
3.2.2	Epipolar Geometry and Horizontal Movement	34
3.3	Stereo Matching and Face Recognition	36
3.3.1	Rectification and Matching Costs	39
3.4	Experiments	42
3.4.1	PIE Pose Variation: 34 Faces	43
3.4.2	PIE Pose Variation: 68 Faces	49
3.4.3	PIE Pose and Illumination Variation	52
3.5	Conclusion	56
4	Face Recognition with Weight Variation	58
4.1	Face Recognition with Weight Variation	58
4.1.1	Experimental Evaluation	60
4.1.2	Discussion	63
5	Face Recognition with Large Pose Variation	67
5.1	Introduction	67
5.2	Stereo Matching with Slant	69
5.3	Dynamic Programming Algorithm	71

5.3.1	The Algorithm	72
5.3.2	Matching Moves	73
5.3.3	Right Occluding Moves	75
5.3.4	Left Occluding Moves	75
5.3.5	Total Cost for Recognition	76
5.4	Experimental Evaluation	76
5.4.1	Pose Variation Experiments	79
5.4.2	Pose+Illumination Variation Experiments	82
5.4.3	Discussion	84
5.5	Conclusions	84
6	Towards Dense, Wide-baseline Stereo under Varying Illumination	85
6.1	Introduction	85
6.2	Previous Work	87
6.3	Stereo Matching Cost	88
6.3.1	Labels	89
6.3.2	Unary Cost	89
6.3.3	Pairwise Cost	91
6.3.4	Labels and GraphCuts	99
6.3.5	Segmentation Cost	100
6.3.6	Discussion	100
6.4	Experimental Evaluation	101
6.4.1	PIE Experiments	102
6.4.2	POVRAY Experiments	103
6.4.3	Outdoor Experiments	104
6.4.4	DAISY Experiments	108
6.4.5	Middlebury Stereo Experiments	108
6.5	Conclusion	108
7	Trainable 3D Recognition Using Stereo Matching	110
7.1	Introduction	110
7.2	Related Work	113
7.3	Stereo Matching Pixel-wise Descriptors	114
7.3.1	Stereo Matching	114
7.3.2	Descriptor Generation	115
7.3.3	Image Representation	116
7.3.4	Epipolar Geometry	117
7.3.5	Classification	117
7.4	Experimental Evaluation	118
7.4.1	CMU PIE Experiments	118
7.4.2	3D Object Categories Dataset Experiments	121
7.5	Conclusion	125

8 Conclusion	126
8.1 Stereo Matching for Face Recognition Across Pose	126
8.2 Dense Wide-baseline Matching with Varying Illumination	127
8.3 Descriptor-based Learning for Face Verification	128
8.4 Onwards	129

List of Tables

2.1	Key aspects of existing methods for face recognition across pose. . . .	15
3.1	Results for pose variation for 34 faces with 3ptSMD. The diagonals are not included in any average. The table layout is the same as [63] and [36]. The global average for this table is 79.8%.	45
3.2	Results for pose variation for 34 faces with 4ptSMD. The diagonals are not included in any average. The table layout is the same as [63] and [36]. The global average is 86.82%	46
3.3	A comparison of our stereo matching distance with other methods across pose.	48
3.4	Comparisons over a slice of the data with the method of Chai et al. [17] and Gross et al. [35]. The gallery pose is c27 and contains 34 faces. The table layout is the same as the one presented in [17]. . . .	49
3.5	Confusion matrix for pose variation for 68 faces with 3ptSMD. The diagonals are not included in any average. The table layout is the same as [63] and [36]. The global average for this table is 74.5%. . . .	50
3.6	Confusion matrix for pose variation for 68 faces with 4ptSMD. The diagonals are not included in any average. The table layout is the same as [63] and [36]. The global average for this table is 82.4% . . .	51
3.7	Summary of the cases where the camera movement is horizontal and when it is not over the experiments with 3ptSMD and 4ptSMD. . . .	52
3.8	Accuracy percentage with pose and illumination variation. The cell format is: (with ambient lights)/(without ambient lights). Three galleries and three probes were used. F: Frontal, S: Side, P: Profile. The table layout is similar to [63]	55
5.1	A comparison of recognition accuracy averaged across pose of our slant-compensated stereo matching distance with other methods. . . .	78
5.2	Pose variation table for 68 faces comparing the use of the stereo matching method of [14] with our slant-compensated method. Cell format: ⟨accuracy Slant SMD⟩ / ⟨accuracy for SMD [14]⟩. Pose pairs are labeled as follows: *: unflipped and pose variation less than 45°, ‡: unflipped and pose variation greater then 45 °and †: flipped pairs. The diagonals are not included in any average. In cells with gray background, the performance gain is significant at 95% (McNemar’s test). The table layout is the same as [63] and [36].	81

5.3	Pose + illumination variation for frontal, side and profile probe table using the stereo matching method of Castillo and Jacobs [14], the method of Romdhani et al. [63] and our slant compensated method. Gray background indicates significant difference between Slant SMD and SMD, the highlighted method is significantly better. F: front, S: side, P: profile.	83
7.1	Decoding of a matching $W = \langle c_1, \dots, c_n \rangle$ into two descriptors D_1 and D_2 of the same length of the scanlines matched.	116
7.2	A comparison of recognition accuracy averaged across pose for our descriptor-based stereo matching distance with other methods on CMU PIE.	120
7.3	Confusion matrix for the 3D object category detection experiment. The overall accuracy over the 70 test objects is 80%.	125

List of Figures

1.1	Example images from the CMU PIE dataset. Observe that no linear transformation can make corresponding boxes have equal size.	3
2.1	Two images from the CMU PIE dataset that show the effect of foreshortening when there is variation in pose.	20
3.1	Our very simplified model of faces.	25
3.2	The circle parameterized by the angle ϕ	27
3.3	Change in disparity relative to the size of the face as a function of θ	28
3.4	Example of our method to compute the epipolar under scaled orthographic projection. For each angle θ we compute the distance perpendicular to it of a fourth point in the two images. We choose the epipolar geometry that has the smallest distance.	32
3.5	Cross-sections with fixed gallery pose for the results presented in Table 3.3. Probe poses marked with * have a vertical misalignment of about 10 degrees with the corresponding gallery pose.	47
3.6	(A comparison of our method with BFS. Gallery pose is frontal (c27) probe poses are as indicated in the x axis, we report the average over the 21 illuminations.	53
4.1	Facial changes as weight variation increases (images shown with permission of subject).	59
4.2	Performance of classifiers by groups of similar relative weight variation.	62
4.3	ROC curve comparing all the non-learning based methods.	63
4.4	Performance of learning-based classifiers by groups of similar relative weight variation.	64
4.5	ROC curve comparing all the evaluated learning based methods. SMD was evaluated on the same testing set for comparison purposes.	65
5.1	A wooden wall with a small patch marked seen from two viewpoints. This example illustrates the critical importance of handling slant correctly.	71
5.2	Gain in performance of Slant SMD compared to SMD (in the 68 face test case), as the angle difference changes. Flipped refers to cases where the azimuthal angles of the poses being compared have different signs, unflipped is when the azimuthal angles have the same sign. All bands show a 90% confidence interval.	78

6.1	Deformation of the matching window under slant and tilt. See text for the values of w_{tl} , w_{tr} , w_{bl} , w_{br}	90
6.2	A wooden wall with a small marked patch as seen from three distinct viewpoints. This example illustrates the critical importance of handling slant and tilt correctly.	91
6.3	A simple example illustrating the inflexion and relatable relation. Relatable pairs are either convex or concave.	93
6.4	An example showing the symmetry of d_{\angle}	94
6.5	$d_{p,q}(f_p, f_q)$ might be computed as $d_{\angle}(f_p, c) + d_{angle}(c, f_q)$ for an appropriate c	94
6.6	Exchanging labels does not produce a symmetric change in geometry, although in the example it does not greatly affect the curvature of an interpolating curve.	95
6.7	Two images from the CMU PIE dataset.	102
6.8	Results of the PIE experiment. Where the x axis is the group and the y axis is the average number of pixels per image labeled incorrectly according to the ground truth.	103
6.9	A left-right pair from our corridor image set across illumination.	104
6.10	Results of the POVRAY experiment. (a) comparison of our slant tilt method with the second order prior method of Woodford, et al. across illumination. (b) comparison of the slant+tilt method with gc+nssd.	105
6.11	Evaluation of our method under illumination variation. The first two rows show original images, the bottom three rows show disparity maps. Red pixels are occluded.	107
6.12	Evaluation of our method under slanted surfaces and a large baseline. Top row: (a) and (b) are the left and right image. (c) and (d) disparity map and image warpings for DAISY (reproduced from [76]) warped images built using the disparity map. Red pixels are occluded. (c) and (d) the disparity map and image warpings for our Slant+Tilt method. In the bottom row, our methods appear to show a slightly different viewpoint due to rectification.	109
7.1	Diagram of our approach. In the rectification step, the images are rectified according to the epipolar geometry. In the description step the descriptors of pixelwise costs are computed. In the backprojection step, the pixelwise costs are transformed back to their original position, by applying the inverse of the rectification transform.	112
7.2	Images from the 13 poses in the CMU PIE dataset.	119
7.3	One example from each class of the 3D Object Category Dataset. Top row: bicycle, car, cellphone and iron. Bottom row: mouse, shoe, stapler and toaster	122

Chapter 1

Introduction

Face recognition is a fundamental problem in computer vision and biometrics. Face recognition has the potential to impact three important application areas. First, improved methods for human face recognition have many applications in security, information retrieval, and HCI. In security, face recognition can be used to control access to sensitive locations (e.g., secure areas of airports). As another example, there is a need for automatic systems in passport applications to alert humans when a new photo does not appear to depict the same person as did a previous one. In information retrieval, a high percentage of photos on the Internet and in personal collections contain faces. The identity and attributes of these faces are a critical element in their effective retrieval. Face identification is also critical to building automatic systems, such as household robots, that can interact smoothly with people. In all these applications, there is a need for systems that can compare images taken in natural imaging conditions, which exceeds the capabilities of current technology.

There are several approaches for separately handling, variation of illumination, pose and expression. However, there are still many unsolved problems when multiple confounding factors occur simultaneously (*unconstrained face recognition*). Progress

in unconstrained face recognition would be important in many applications, for example: surveillance, security, the analysis of personal photos and other domains in which we cannot control the conditions under which the images are taken.

There has been a lot of progress in the case of images taken under controlled conditions [84]. There are many approaches for handling, variation of illumination and expression. There are also several approaches to handling pose variation [63, 33, 35, 17]. However, there is still a lot of room for improvement. Progress in unconstrained face recognition would be important in many applications, for example: surveillance, security, the analysis of personal photos and other domains in which we cannot control the conditions under which the images are taken.

Existing systems achieve excellent results when images are taken under controlled conditions, so that there is no variation in viewing conditions. Recently, there has been a good deal of work on recognition in the case of variations in viewing conditions that occur over a short period of time, such as variations in pose or lighting. Variations that occur over longer periods of time (such as aging and weight gain) have proven harder to study.

1.1 Stereo for Face Recognition

Correspondence seems crucial to produce meaningful image comparisons. The importance of good correspondences is even greater in the case of face recognition across pose. Standard systems often align the eyes or a few other features, using translation, similarity transformations, or perhaps affine transformations. However, when the pose varies these can still result in fairly significant misalignments in other parts of the face. Observe, for example, that in Figure 1.1 no linear transformation

can make corresponding boxes have equal size, because a linear transformation can only linearly scale their size.

To handle this situation, we use stereo matching. This allows for arbitrary, one-to-one continuous transformations between images, along with possible occlusions, while maintaining an epipolar constraint. We show that the greater generality provided by stereo matching, which efficiently computes dense correspondences, may be necessary for effective face recognition across pose.

The purpose of stereo matching is to compute correspondences between scan lines of pixels in images. Correct correspondences can be many-to-one and can involve occlusions. This means that situations like the one presented in Figure 1.1 can be handled by stereo matching.

In the process of computing the correspondences between scan lines in two images a *stereo matching* cost is optimized, which reflects how well the two images match. We can use the stereo matching cost as a measure of similarity between two face images.

Note that we are not interested in performing 3-D reconstruction, which is the most common purpose of stereo matching. In reconstruction the stereo matching costs are discarded and the correspondences are used along with geometric infor-

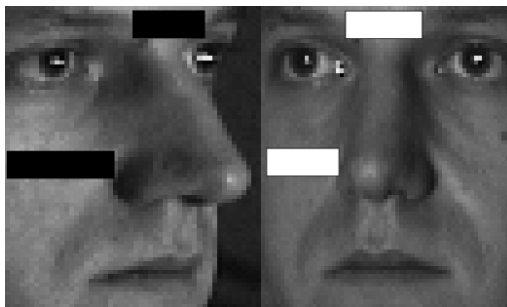


Figure 1.1: Example images from the CMU PIE dataset. Observe that no linear transformation can make corresponding boxes have equal size.

mation about the camera layout to compute a 3-D model of the world. We have no use for the correspondences except to compute the stereo matching costs. We are therefore unaffected by some of the difficulties that make it hard to avoid artifacts in stereo reconstruction. For example, ambiguities frequently arise when different correspondences produce similar costs; in this case selecting the correct correspondence is essential for reconstruction, but not very important for judging the similarity of two images.

Prior to stereo matching, we need to estimate the epipolar geometry. In almost all applications of face recognition, the size of the face is small relative to its distance to the camera. Therefore we can approximate the projection of the face to the camera using scaled-orthographic projection (weak perspective). Under scaled-orthographic projection all epipolar lines are parallel to each other (the epipole is at infinity). This simplifies the problem of determining the epipolar geometry.

We propose two methods. One method uses four feature points to estimate the epipolar geometry of the two faces. The images are then rectified, and the similarity score is computed by adding the stereo matching cost of every row of the rectified images. The method works with general camera movement under the (very reasonable) assumption of scaled orthographic projection. We also study a specific case in which the camera is at the same height as the eyes of an upright subject. In this case, the epipolar lines are parallel to the lines that connect the two eyes. In this case we can determine epipolar geometry using only three points.

Putting these steps together, we have the following, remarkably simple algorithm:

- Prior to recognition, build a gallery of 2D images of faces, each with three to four landmark points specified.

- Given a 2D probe image, find three to four corresponding landmark points.
- Compare the probe to each gallery image as follows:
 - Using landmark points, rectify the probe and gallery image.
 - Run a stereo algorithm on the image pair, using the enhancements described in Section 3.3. Discard the correspondences and use the matching cost as a measure of image similarity.
- Identify the probe with the gallery image that produces the lowest matching cost.

We will show that this method works very well even for large viewpoint changes. We evaluate our method using the CMU PIE dataset. Our results show that with pose variation at constant illumination our method is more accurate than previous methods due to Gross et al. [35], Chai, et al. [17] and Romdhani et al. [63]. While our method is designed to only handle pose variation, we also test it with pose and illumination variation to verify that our method does not fall apart in such a setup. Surprisingly, our method is more accurate than the method of Gross et al. [33], which is designed to handle lighting variation, though it is not as accurate as the method of Romdhani, et al. [63].

1.2 Improving on Stereo for Face Recognition

After finishing our initial work on face recognition across pose using stereo we realized several things:

- This approach to face recognition stresses stereo matching algorithms significantly. When comparing faces taken from very different viewpoints, one

essentially must perform stereo matching with a very wide baseline. While a great deal of progress has been made in wide baseline stereo [49], these approaches generally do not produce a cost based on dense correspondences that is appropriate for image comparison and face recognition.

- Although large changes in pose do create significant occlusions in a face, they generally do not affect the monotonicity of correct matches. Even when matching a frontal view of someone to her profile, we can establish a continuous matching over one half of the face.
- When using stereo for recognition but not for reconstruction we have other demands than people using stereo for reconstruction. People using stereo for reconstruction care about the quality of the disparity map and the quality of the correspondences around depth discontinuities and in smooth untextured regions, while when using it in recognition we only care that when comparing faces of different people stereo matching gives a high cost and when comparing faces of the same person it gives a low cost.

To address these findings, we developed a dynamic programming-based stereo algorithm that might be unsuitable for wide-baseline matching of more general scenes. In doing so we found that in wide-baseline stereo slant and tilt affect the appearance of an object. This creates a chicken-and-egg problem in which it is difficult to find the right match for image points without knowing the slant and tilt, but one needs correspondences to determine the slant and tilt. However, pose variation in faces tends to produce foreshortening primarily in the direction of the epipolar lines. We show that this allows us to use dynamic programming to solve for the main component of foreshortening at the same time that we find correspondences.

Our dynamic programming algorithm that accounts for the effect of slant works very well in face recognition across pose, in particular significantly better than our previous stereo method.

1.3 Advancing Towards Dense Wide-baseline Stereo Under Varying Illumination

We then continued studying the problem seen from a classical stereo point of view. We realized that we could extend our insights in an MRF (Markov Random Fields) formulation to stereo matching. In our formulation, pixels are labeled according to their disparity and relative slant and tilt. This allows us to compare pixels in different images using windows that are rectified to allow for changes in window shape due to viewpoint change. A key contribution is the proposal of a new pairwise cost function that measures the consistency between neighboring labels. This pairwise cost is a metric, allowing us to use Graphcuts to optimize the resulting cost.

Representing disparity, slant and tilt leads to a potentially huge label set. However, each label represents a planar surface; consequently a relatively modest number of labels are needed to accurately approximate any given scene. We exploit this using an algorithm in which we incrementally add labels as needed, so that our Graphcuts problem remains manageable.

Our cost can be adapted to any stereo matching method that uses windows or regions when comparing pixels. We experiment using a very simple but popular approach of comparing windows by the sum of square differences between their normalized intensities. Our experiments focus on showing that stereo matching

using slant and tilt can provide a substantial improvement over matching that uses only fixed sized windows, or over pixel based matching. We do this using our own new dataset of outdoor images taken with wide baselines and lighting variation, using face images with varying pose and lighting, from the CMU PIE [70] dataset and also using the wide baseline images from the DAISY [76] dataset. We also show experiments on the standard Middlebury data set[67].

1.4 Organization

This dissertation is organized as follows: Chapter 2 is about related work. Chapter 3 introduces the usage of stereo matching for face recognition and it is mostly based on the following two papers:

- Carlos D. Castillo and David W. Jacobs, "Using Stereo Matching with General Epipolar Geometry for 2-D Face Recognition Across Pose" , IEEE TPAMI, December 2009.
- Carlos D. Castillo and David W. Jacobs, "Using Stereo Matching for 2-D Face Recognition Across Pose", IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), 2007.

Chapter 4 presents our work using stereo matching as a method to recognize faces in the presence of weight variation; this part of the work emphasizes the importance of finding correspondences and shows the feasibility of using stereo, even when the deformation is not rigid (such as weight variation and slight variation in expression).

Chapter 5 is about using stereo matching for face recognition under large pose variation and it is based on the following paper:

- Carlos D. Castillo and David W. Jacobs, "Wide-Baseline Stereo for Face Recognition with Large Pose Variation" , IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), 2011.

Chapter 6 is about advancing towards dense, wide-baseline stereo under varying illumination in an MRF-based stereo setup, it is based on the following paper currently under review:

- Carlos D. Castillo and David W. Jacobs, "Towards Dense, Wide-baseline Stereo under Varying Illumination" , IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), 2012. Under review.

Chapter 7 is about extending our work on stereo-based image comparison to include the possibility of learning from stereo-based descriptors and it is mostly based on the following paper:

- Carlos D. Castillo and David W. Jacobs, "Trainable 3D Recognition Using Stereo Matching" , 3dRR (Workshop on 3D Representation and Recognition held at ICCV), 2011.

Finally, Chapter 8 presents our conclusions.

Chapter 2

Related Work

Face recognition is a fundamental problem in computer vision. It has been widely studied for the past 30 years. There has been significant of progress in this research area, see [84] for an excellent survey.

In the past few years, interest in face recognition in unconstrained settings has grown dramatically. Unconstrained is understood to mean simultaneous variation in illumination, pose, expression, time and weight. To systematically study these variations, they have been separated into tractable groups, such as variation of pose and illumination, expression and pose, aging, etc. There are several variations for which a much progress has been made and there are other variations for which there is a lot of room for improvement.

2.1 Face Recognition Across Pose

Zhao et al. [84] review the vast literature on face recognition. Although the bulk of this work assumes fixed pose, there have been a number of approaches that do address the problem of pose variations.

Correspondences are fundamental for face recognition across pose. Many of these methods use some 3-D knowledge of faces to compensate for pose. In this case, obtaining correspondences becomes an operation of aligning the 3-D model to 2-D images: morphable model fitting, 3-D rigid transformations, and sampling images from a 3-D model have been proposed. Other methods only use image information (i.e., they don't use 3-D knowledge). In this case obtaining correspondences becomes a 2-D matching problem: optical flow, estimating the light-field of the object, and a wide variety of patch-based methods have been proposed. Typically, these approaches all rely on some initial manual correspondences. It is expected that if a method obtains good correspondences, it should obtain effective performance at face recognition across pose. Table 2.1 presents a summary of existing methods of face recognition across pose.

Historically, many approaches compensate for some 2-D deformations in matching, which may partially compensate for the effects of pose. A notable example is the work of Wiskott et al. [81]. This work was among the first to present a face recognition method that was robust to alignment issues. They developed a method called Elastic Bunch Graph Matching (EBGM). The comparison function used Gabor jets at manually clicked feature points, and geometric information of distances between the feature points. Correspondences were obtained for the feature points only.

One of the first methods to study face recognition across pose was proposed by Beymer and Poggio [7]. In their work they generated 2-D virtual views from a single image per person using prior knowledge of the object class (in particular symmetry and prototypical objects of the same class) using optical flow. Once the virtual view had been generated the images were compared. Our method is similar to theirs in

the sense that both are decidedly 2-D and stress the importance of finding good correspondences. In their approach the correspondences are obtained using optical flow between the two facial images.

Blanz and Vetter [9] use laser scans of 200 subjects to build a general 3-D morphable model of three-dimensional faces. Then, with the aid of manually selected features, they fit this model to images. The parameters of the fit to two different images can be compared to perform recognition. In their experiments they show strong results for a subset of the poses in the PIE database. The work of Romdhani et al. [63] also focuses on 3-D morphable models. In this work shape and texture parameters of a 3-D morphable model are recovered from a single image. They present exhaustive results of experiments with pose variations for the PIE dataset and show strong results (these are the best prior results we are aware of with pose variation). In these methods the correspondences are obtained by fitting the 3-D morphable model to the 2-D images. These type of methods solve a very difficult intermediate problem (fitting or inverse rendering) which is useful for graphics, but may not be needed for recognition.

Basri and Jacobs [4] use a 3-D model to generate a low dimensional subspace containing all the images that an object can produce under lighting variation. Pose is determined using manually selected point features. Correspondences are obtained by computing a 3-D rigid transformation that aligns the features of a 3-D model with the corresponding features of the 2-D images.

In Georghiades et al. [29] a 3-D model is computed for each person using a gallery containing a number of images per subject taken with controlled illumination at a constant pose. Pose variation is handled by sampling the set of possible poses, and building a 2-D model for each one. They evaluate their method using the Yale

Face Database B. Correspondences with 2-D images are obtained by sampling the individual 3-D head model.

In Gross et al. [33] two appearance-based algorithms for face recognition across pose and illumination are presented. One of them is called eigen light-fields. At the core of the method is the *plenoptic function* or light field. To use this concept, all of the pixels of the various images are used to estimate the (eigen) light-field of the object. Correspondences are obtained by computing the light-field angles using the camera intrinsics and the relative orientation of the camera to the object (which are assumed to be known). They evaluate their results using the CMU PIE dataset [70]. In its assumptions, recognizing faces across general unknown poses, this method is the most similar to ours. However our approach is simpler and our results are better.

The other method presented in Gross et al. [33] is called Bayesian Face Subregions (BFS). The algorithm models the appearance changes of the different face regions in a probabilistic framework. Using probability distributions for similarity values of face subregions, the method computes the likelihood of probe and gallery images coming from the same subject. The method is designed to handle the case of simultaneous variation in pose and illumination. In this patch-based method, correspondences are computed trivially on a quadrilateral grid that includes the two eyes and the mouth as edges.

There have been several recent approaches to face recognition across pose that are based on patches. In Chai et al. [17], the authors present a learning, patch-based rectification method based on locally linear regression. Given a non-frontal facial image, the method provides a prediction strategy to generate the frontal view. In their experiments, the method compares well to other recent methods on the PIE dataset. Lucey and Chen [48] present a patch-based algorithm for face recogni-

tion across pose of sparsely registered images (4 manually selected points). Closely related, the work of Ashraf et al. [3] presents a new method to discover viewpoint-induced spatial deformations for general patch-based methods of face recognition across pose.

There have been many methods recently proposed that use tools from numerical linear algebra to handle pose variation. Examples of this work are Prince, et al. [61] that use tied factor analysis and Sharma and Jacobs [69] that use partial least squares. These methods exhibit great performance, however it is unclear if they can be extended to generalize to previously unseen poses.

All the methods previously mentioned in this section use intensity images of the face. This type of face recognition, based on 2-D images constitutes the vast majority of face recognition research. There is, however, a significant amount of work done acquiring, matching and performing recognition using 3-D reconstructions of faces (see [11] for a survey).

While progress has been made in handling pose variations, significant challenges remain. For this problem, current methods have substantially worse performance than when pose is fixed between the probe and gallery. In addition, many methods for handling pose variation require substantially more computation than other methods, and can be very slow. This is in part because the process of finding a correspondence between the probe and the gallery requires expensive optimization processes.

Table 2.1: Key aspects of existing methods for face recognition across pose.

Method	Type	Correspondences	# of manually specified points
Wiskott et al.	2-D	Jets only at points with manually specified correspondences	4-7
Beymer and Poggio	2-D	Optical flow	4-6
Blanz and Vetter	3-D general model	3-D model fitting	10-20
Romdhani et al.	3-D general model	3-D model fitting plus extensions	10-15
Basri and Jacobs	3-D person-specific model	3-D rigid transformation	5
Gheorgiades et al.	3-D person-specific model	Sampling from a built 3-D model	Requires training and test images in the same pose
Gross et al. (ELF)	2-D	Computing the eigenlightfield, known camera geometry	3/40+
Gross et al. (BFS)	2-D	Patches, sampled uniformly on the central region of the face	3
Chai et al.	2-D	Rectification through locally-linear regression	5
Lucey and Chen	2-D	Patches, learning patch dependency	4
Ashraf et al.	2-D	Patches, learning the spatial deformation of the patches	4

2.2 Face Recognition Across Illumination

In addition to pose we also consider work related to lighting, as our work addresses lighting. We focus on papers that illustrate 2-D methods and that focus on representations that are robust to illumination change.

Adini et al. [2] present a great study of the sensitivity of several representations of the facial images to variations in illumination in face recognition and illustrate the significance of this issue.

Many representations have been proposed that are robust to variation in illumination. One paper that studied many representations and proposed several alternatives with increasing level of complexity is the work of Chen, et al. [20], including comparing the direction of gradient. Later on Gopalan and Jacobs studied the performance of several state of the art representations and their potential for integration [31].

Osadchy et al. [58] have shown that many commonly used illumination invariant representations are equivalent with respect to their expressive power and their robustness, in particular they show that the direction of gradient and normalized correlation over small windows are exactly the same when the intensity change inside a window is assumed to be linear.

Recently several powerful image representations for face recognition under varying illumination have been proposed, examples of which are the self quotient image (SQI) [80] which uses the quotient of an image and its diffused version as representation and the work of Tan and Triggs [74] which uses local binary patterns to normalize and represent the image.

The key to this section is to point out that most if not all successful representations for face recognition with illumination change use relatively large support regions to normalize the image for later comparison, this fact will turn out to be quite important for the work presented here.

2.3 Unconstrained Face Recognition

In the past few years, there has been great interest in face recognition in unconstrained settings. Consequently, researchers have produced new datasets of

images acquired in unconstrained environments. One notable example of such a dataset is Labeled Faces in the Wild (LFW [39]). This is a huge collection of those images from the news in which the Viola and Jones [79] detector is able to find faces.

Kumar et al. [42] present a set of methods for face recognition using high level describable visual features (such as blonde, brunette, smiling, has glasses, mouth open, eyes open, young, middle aged, senior, etc.). In their work they present an alternative to LFW, which is both more difficult and larger. The new dataset is called PubFig which is a dataset of images of public figures for which the authors were able to obtain many images (more than 50 per individual).

Phillips, et al. present a dataset called Good, Bad and Ugly (GBU) which includes three datasets that go from controlled (Good) to unconstrained (Ugly), and illustrates how the recognition rate decreases as the imaging conditions become more and more unconstrained. In this particular dataset the difficulty mostly stems from lighting and expression and not so much from pose difficulties. Additionally, the selection of which faces are good, bad and ugly stems from the performance of an ensemble of the top performing methods from the FRVT (Face Recognition Vendor Test).

Ramanathan and Chellappa [62] studied the problem of matching face images taken years apart, and proposed an adaptation of the probabilistic eigenspace framework[53]. Ling et al. [46] also studied this problem and proposed an algorithm based on learning facial differences that are described using a gradient orientation pyramid (GOP).

There are also a wide variety of datasets that provide systematic variation of one or several confounding factors in face recognition. One such database obtained in an unconstrained setting is the BioID dataset [40]. The BioID dataset

aims to capture significant variability in pose, lighting and expression. Images are captured in a realistic setting, for example in a home environment. There are also many datasets that provide systematic variation of confounding factors obtained in controlled conditions. One of the most widely used is CMU-PIE [70] which provides systematic variation over pose, illumination and expression for 68 individuals. The Face Recognition Grand Challenge (FRGC) [59] presents a six-experiment challenge problem along with a dataset of 50,000 images. The images in the dataset are collected both in controlled and uncontrolled settings.

Algorithmically, the key methods to handle unconstrained face recognition can mostly be categorized as follows:

1. Descriptor: most methods use some type of representation which can handle illumination and expression (Local Binary Patterns is a great example of this, see for example Wolf et al. [82], SFI and Tan and Triggs are another example of an effective representation)
2. Learning: once the description of the images has been computed a learning mechanism is invoked. Examples of learning methods are: ITML [37], SVM [82] and Partial Least Squares [68], background samples using one shot learning [73], etc.

In the particular problem of unconstrained face recognition, which started to receive attention in the past five years, great progress has been made, but there are still long ways to go. For example, on LFW for verification the equal error rate (EER) has gone from 65% to 90% but in controlled conditions we can obtain equal error rates of more than 98%.

2.4 Stereo Matching with Slant

Our approach makes use of window-based, dense stereo matching. That is, given a left and a right image, we want to assign to each pixel a disparity d so that every point (x_l, y_l) on the left image matches a point $(x_l + d, y)$ on the right image. Specifically, we build on the method of Criminisi et al. [22], which compares windows using an approximation to normalized correlation. This has been shown to be very effective for face recognition with pose and lighting variation. Other representations have been suggested in face recognition to handle lighting variation [31]; we do not consider these directly, but they generally will suffer from the effects of pose variation in ways that are similar to window-based methods. Wide baseline stereo has been addressed with other approaches, such as feature-based matching. However, these approaches seem less suitable for image comparison and face recognition because, by design, they do not evaluate a cost that accounts for the entire image (eg., Matas et al. [49]).

One of the key issues in dense, wide baseline stereo is the considerable difference in foreshortening that can occur when a face is viewed from different viewpoints. This effect can be seen in Figure 2.1. This issue is elegantly described by [44, 23]. Following Li and Zucker [44] we characterize a plane on which a point $p = (u, v)$ falls with disparity d as either:

- fronto-parallel ($\frac{\partial d}{\partial u} \approx 0, \frac{\partial d}{\partial v} \approx 0$),
- *slanted* ($\frac{\partial d}{\partial v} \approx 0, \|\frac{\partial d}{\partial u}\| \gg 0$),
- *tilted* ($(\|\frac{\partial d}{\partial v}\| \gg 0, \frac{\partial d}{\partial u} \approx 0)$ or,
- otherwise, in general configuration.

They show that when a surface is fronto-parallel, using fixed sized windows is valid, but otherwise matching windows will vary significantly in shape and size, which can produce significant errors. With wide-baseline matching these effects become significantly exacerbated.



Figure 2.1: Two images from the CMU PIE dataset that show the effect of foreshortening when there is variation in pose.

The work of Criminisi, et al. elegantly handles slant in the matching produced between pixels, by allowing many-to-one matchings. So when a slanted surface produces a different number of pixels in the two images, the correct correspondences can be found. However, their method does not account for changes in the size and shapes of the windows being matched, and when matching slanted surfaces this leads to systematic errors. The work of Li and Zucker [44] and Devernay and Faugeras [23] handles slant and tilt in matching and in the windows, but they use an iterative algorithm that assumes that correct correspondences can be initialized without accounting for slant and tilt. These methods seem most appropriate for small baselines. In particular, [44] and [23] focus on accounting for slant and tilt to produce accurate subpixel estimates of disparity in situations in which normal stereo matching might produce accurate pixel-wise correspondences. The method of Birchfield and Tomasi [8] can handle arbitrary slant but since it matches individual pixel intensities, it will be very sensitive to lighting variation. In general several

existing methods study slant and tilt for stereo but are really not intended for wide-baseline situations.

Most of stereo matching assumes two images of the same scene taken at the same instant of time. We would like to study the problem of stereo matching in the presence of illumination change; these conditions imply that the images were not taken at the same instant of time. Many methods (see Ogale and Aloimonos [55], for example) have provisions to handle small variations in illumination to compensate for photometric issues. On the other hand, we are interested in dense stereo with major changes in viewpoint and the interaction of changes in illumination and viewpoint when matching very slanted surfaces.

2.5 MRF Stereo

We also study a Markov Random Field (MRF) formulation for stereo. Many recent papers have proposed new, effective optimization algorithms for use in stereo matching using MRFs (eg., belief propagation [72] and QPBO-I [64]). When pairwise costs between pixels obey a regularity, or metric constraint, graph cuts-based methods [12] have proven extremely effective and efficient, and we use this approach.

Other important recent advances in stereo include the use of segmentation (eg., [41]). While of great interest, these approaches are largely orthogonal, and potentially complementary to our work.

Several authors [23, 44] discuss the effect that slant and tilt have on window-based matching. When the baseline is not wide, the goal of these approaches is to compensate for small changes in foreshortening, (eg., [56]) or to use the subtle effects of foreshortening to perform matching with subpixel accuracy. [45] propose

a belief propagation-based framework for stereo matching in the presence of slanted and curved surfaces. Also, [15] propose a dynamic programming method that accounts for slant (but not tilt) in wide baseline matching of faces. Building on this understanding of slant and tilt, our goal has been to construct metric cost functions that allow us to use graph cuts to efficiently perform matching in wide baseline settings.

We represent the relative slant and tilt of a surface at each pixel, in terms of the horizontal and vertical changes in disparity. Previous approaches have also represented surface orientations in stereo matching. [10], for example, label patches of images with planar surfaces or b-splines in the scene. One minor advantage of our disparity-based representation is that it is suitable for situations in which the epipolar lines of an image pair have been determined, but the magnitude of the baseline is unknown.

Chapter 3

Stereo for Face Recognition Across Pose

In this chapter we will present a first pass at using stereo for face recognition across pose using an existing, off-the-shelf algorithm for stereo matching. Surprisingly this turns out to be an effective method face recognition across pose.

3.1 Analysis of Stereo Matching for Face Recognition

Most work in image-based recognition aligns regions to be matched with a low-dimensional transformation, such as translation, or a similarity or affine transformation. Instead, we use stereo matching. When we enforce the ordering constraint, this allows for arbitrary, one-to-one continuous transformations between images, along with possible occlusions, while maintaining an epipolar constraint. In this section we show that the greater generality afforded by stereo matching may be necessary for face recognition, and that stereo matching will not be too sensitive to noise in determining the epipolar lines.

We illustrate this using a *very* simplified model of faces, in which we calculate the disparity maps that will correctly match two images. We do not attempt to accurately capture face shape in this example. Rather, we just provide a coarse demonstration of the disparity variation that can occur under viewing conditions similar to those that typically occur in face recognition.

1. We model the face as a cylinder. Perturbations to this model, such as adding a nose, can be handled fairly easily.
2. We assume the face is viewed by two cameras with image planes that are rectified to be perpendicular to the z axis and that the cylinder axis is the y axis. This is roughly the situation when an upright person photographs another upright person. For simplicity, we will assume that the cylinder lies on the z axis, that the camera focal points lie on the x axis at points symmetric about the z axis (see Figure 3.1). We call the left and right focal points f_l and f_r respectively.
3. We assume that the distance from the camera to the person is much bigger than the radius of the cylinder that represents the person. Specifically, we assume that vectors from the camera focal point to any location on a horizontal cross section of the cylinder have the same direction. If we imagine that the cylinder (face) has a radius of three inches, and the distance from the camera to the face is 8 feet, we can calculate that a vector from the focal point to the center of a cross-section of the cylinder will be within 5.5 degrees of a vector to any point on the cylinder cross section, so this approximation is not too bad.

These assumptions simplify our presentation, which could be readily extended to other settings.

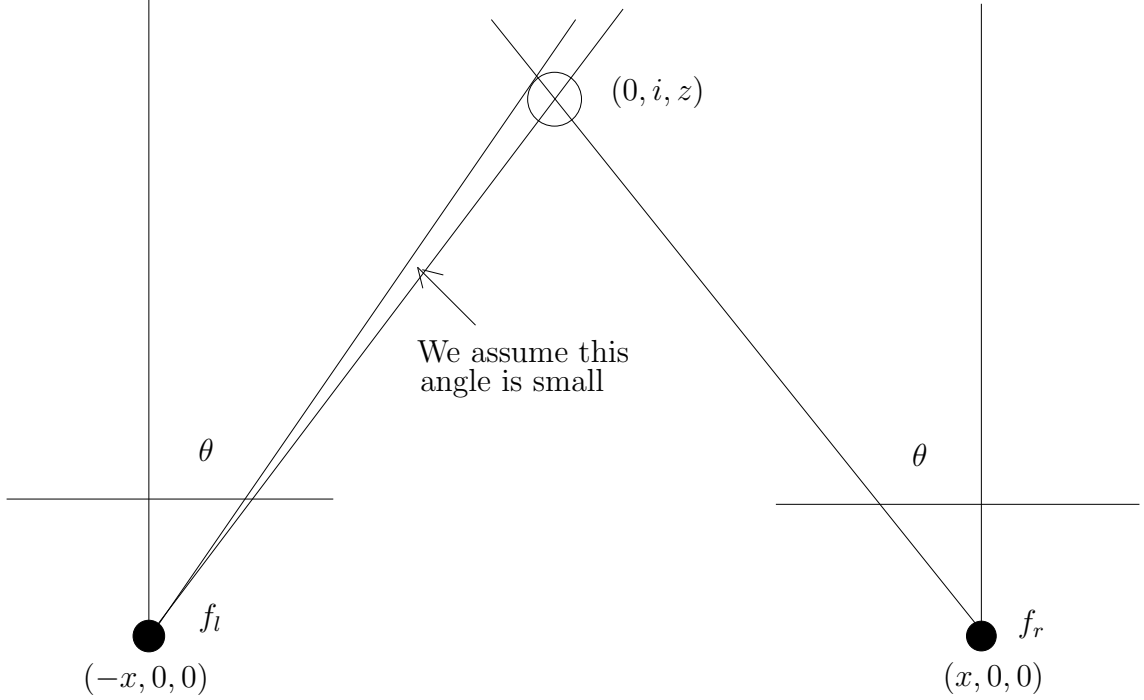


Figure 3.1: Our very simplified model of faces.

We will analyze disparities on the $y = 0$ plane. Given these assumptions, each camera will see half of a circular cross-section. They will not see exactly the same half-circle, however, as there will be some occlusion. Without loss of generality assume the radius of the circle is 1. We will denote the angle between the z axis and a vector from f_l to the cylinder by θ . The corresponding angle for the right camera will then be $-\theta$. Define l_1 and l_2 to be two points on the circle, such that the tangent lines to the circle at l_1 and l_2 pass through f_l . That is, l_1 and l_2 are the first and last points on the circle that are visible in the left image. Define L to be the line connecting l_1 and l_2 . We can similarly define r_1 and r_2 for the right image. So, for example, the region of the circle between r_1 and l_2 is visible in both images.

Note that every line connecting f_l to L intersects the circle in a single point that will be visible in the left camera. So one way to determine the image of the circle in the left camera is to project the visible half-circle onto L using these lines,

and then to consider how L is projected onto the left camera. Because we assume the cylinder is small relative to its distance to the camera, we can approximate the projection of L into the left camera using scaled-orthographic projection. Without loss of generality we can normalize the left image so that the width of the circle's projection is 1 (this is in image units, which may differ from 3D units), and the x coordinate of the image of l_1 is 0. This is illustrated in Figure 3.2.

We can parameterize points on the circle by the angle ϕ , which we take relative to l_2 (see Figure 3.2). Consider some such point p . We can determine the location of p in the left image, by considering the line through p and f_l . The point where this line intersects L , call it P_l , will appear in the same image location as p . Define the distance from P_l to l_1 to be $d(l_1, P_l)$. Then the x coordinate of p in the left image is $d(l_1, P_l)/2 = (1 + \cos \phi)/2$. Similarly, its position in the right image will be $(1 - \cos(\pi - 2\theta - \phi))/2$. If we define the disparity, d , in a matched point to be its x coordinate in the left image minus the x coordinate in the right; we get:

$$d = (\cos \phi + \cos(\pi - 2\theta - \phi))/2 \tag{3.1}$$

It is straightforward to show that disparity is minimized by $\phi = 0$ or $\phi = \pi - 2\theta$, which are the furthest points visible in both cameras, and maximized by $\phi = (\pi - 2\theta)/2$, which corresponds to the point closest to the cameras.

We are interested in the variation between the minimum and maximum disparity values, Δd . We have:

$$\Delta d = \cos\left(\frac{\pi}{2} - \theta\right) - \frac{1}{2} - \frac{\cos(\pi - 2\theta)}{2} \tag{3.2}$$

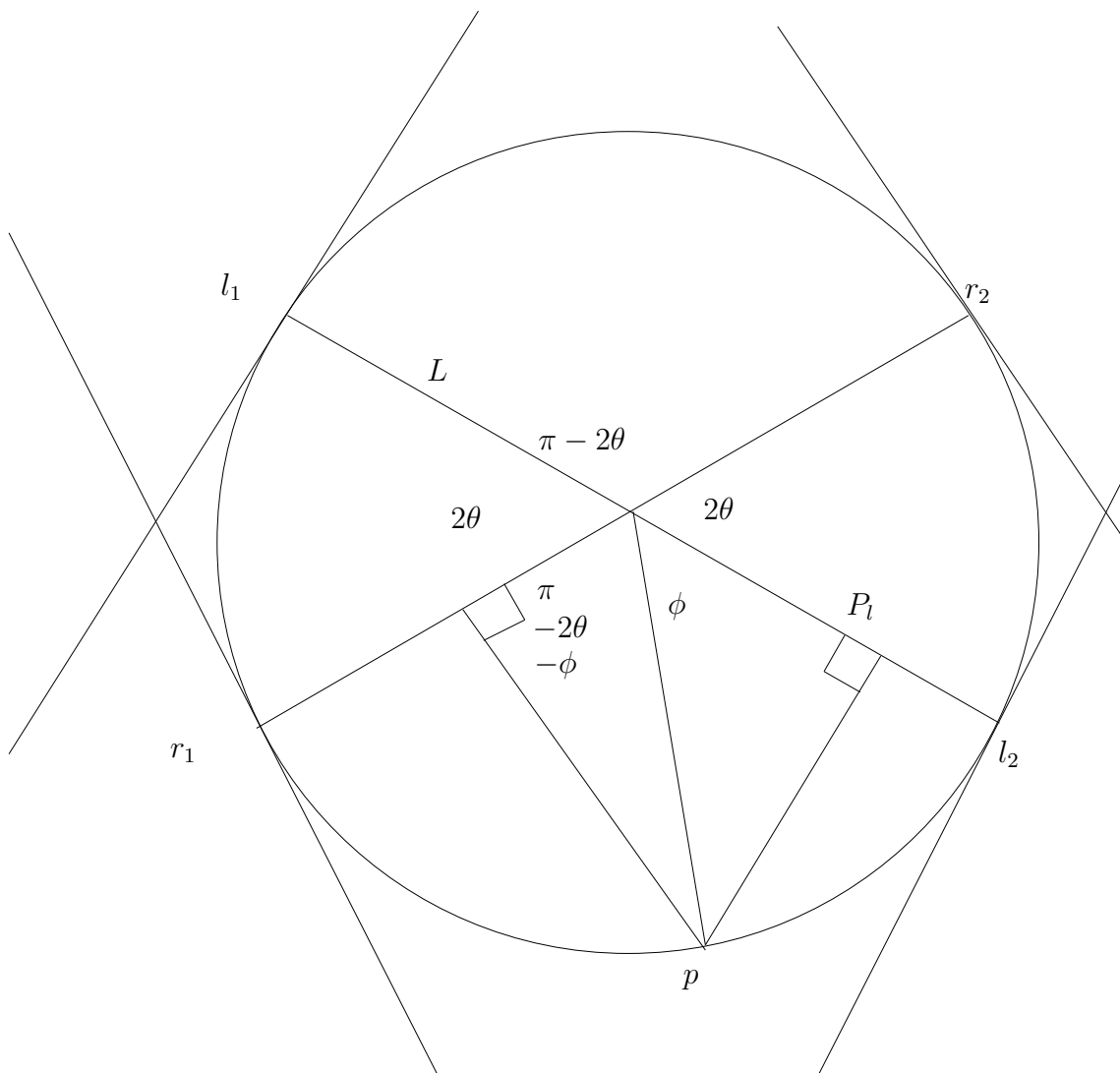


Figure 3.2: The circle parameterized by the angle ϕ .

This is maximized for $\theta = \pi/6$, when $\Delta d = 1/4$. Figure 3.3 shows how the maximum change in disparity varies with θ . In Figure 3.3, we can see that for a large range of θ , disparity changes quite a bit within the image.

From this analysis, we can see that for a cylinder, disparity in an image can vary by as much as 1/4 of the apparent width of the cylinder, and frequently varies substantially. These variations in disparity cannot be accounted for by aligning the images with a linear transformation, since linear transformations can only create linear disparity maps. In contrast, the disparity map for this cylinder is highly non-linear, since the smallest disparity is at the two ends of the image, and the greatest disparity occurs in the middle of the image. In fact, in scenarios such as the one described here, because of the symmetry of the viewing conditions, we can demonstrate that the optimal linear transformation to align the two images will simply be the identity transformation, which does not account for any of these variations in disparity. Note that the amount of disparity is independent of the

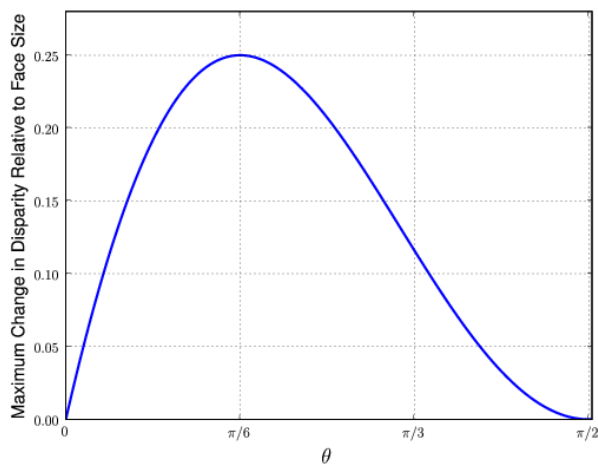


Figure 3.3: Change in disparity relative to the size of the face as a function of θ .

distance from the cameras to the face, because we measure disparity relative to the apparent size of the face.

Ideally, one should determine the epipolar geometry prior to matching two faces. However, in many cases, images result from an upright photographer taking a picture of an upright subject. This results in epipolar lines that are approximately horizontal. If we align the eyes in two photographs, this will align corresponding horizontal epipolar lines. However, error will result when epipolar lines are not purely horizontal. To get a sense of the possible magnitude of this error, we analyze a simple example.

Consider the case in which we take two pictures of a face that is five feet high, at a distance of eight feet. But suppose that the disparity is vertical instead of horizontal, because one photograph is taken from a height of five feet, and the second is taken from a height of six feet. Vertical disparity will be zero at the eyes, which are aligned, and will be maximized at the point that is closest to the cameras, the tip of the nose. If we assume that the nose is about one inch long, then using similar triangles we can determine that it appears at the same image location as a point $1/8$ of an inch below the nose, in the second image. For a face that is six inches long, the vertical disparity will therefore be about 2% of the height of the face in the image. This error is small compared to the variations of up to 25% in horizontal disparity that can arise in the situation we analyze above. Of course, this is just an illustrative example; the error introduced by mis-estimation of the epipolar lines will depend in practice on the viewing conditions typical in a specific application. Our example simply makes the point that in some common settings, this error will be quite small, while stereo matching can compensate for correspondence errors that will be large.

3.2 Alignment

In order to perform stereo matching we first need to know the epipolar geometry. In the most general case this requires eight corresponding points. We can reduce this by assuming that images are generated by scaled orthographic projection. This model is valid when the average variation of the depth of the object along the line of sight is small compared to the distance of the camera to the object and the field of view is small as is generally the case with facial images. Note that, as shown in Section 3.1, even with scaled orthographic projection there can be considerable variation in disparity between two images.

To begin, consider the case of two images generated with orthographic projection. Orthographic projection occurs with a perspective camera model when the focal point is at infinity. The *baseline*, which connects the two focal points, is therefore a line at infinity. The *epipole* of each image, then, is a point where this line at infinity intersects the image plane. This means that the epipoles are points at infinity in each image plane. The *epipolar lines* in each image therefore intersect at a point at infinity, meaning that they are parallel. If we also allow for scaling in each image, this may alter the distance between corresponding epipolar lines, but will not affect the fact that they are parallel.

As we will demonstrate, we can calculate the epipolar geometry under the scaled orthographic model using four feature points. We will not focus our attention on how these points can be obtained; in our experiments we specify them by hand. Some applications involving off-line recognition may use such hand clicked points directly. At the same time there is a lot of work on automatic detection of facial features [30, 38, 19, 65]. By reducing the number of points needed for recognition,

we can make it easier to use these detectors to build fully automatic recognition systems.

3.2.1 Epipolar Geometry under Scaled Orthographic Projection

We now want to consider arbitrary viewpoint changes, still using scaled orthographic projection. Under scaled orthographic projection the epipolar geometry can be characterized as a tuple: (θ, γ, s, t) . θ is the angle of the epipolar lines in the first image. γ is the angle of the epipolar lines on the second image. s is the relative scale; that is, scaling the second image by s will cause the distance between two epipolar lines in the second image to match the distance between corresponding lines in the first image. Finally, t is the translation perpendicular to the epipolar lines needed to align corresponding lines.

Solving for this type of epipolar geometry requires four corresponding points. We formulate this by encoding the three variables relating to the second image, (γ, s, t) , as a similarity transformation, with the added constraint that the translation must be perpendicular to the epipolar lines. Given corresponding points in the two images, this similarity transformation must transform each point in the second image onto a line in the first image that passes through the corresponding point, at an angle θ . This yields a constraint for each point of the form:

$$(P_{2x}^i, P_{2y}^i, 1) \begin{pmatrix} a & -b & 0 \\ b & a & 0 \\ T_x & T_y & 1 \end{pmatrix} \begin{pmatrix} l_{1x}^i \\ l_{1y}^i \\ l_{1c}^i \end{pmatrix} = 0 \quad (3.3)$$

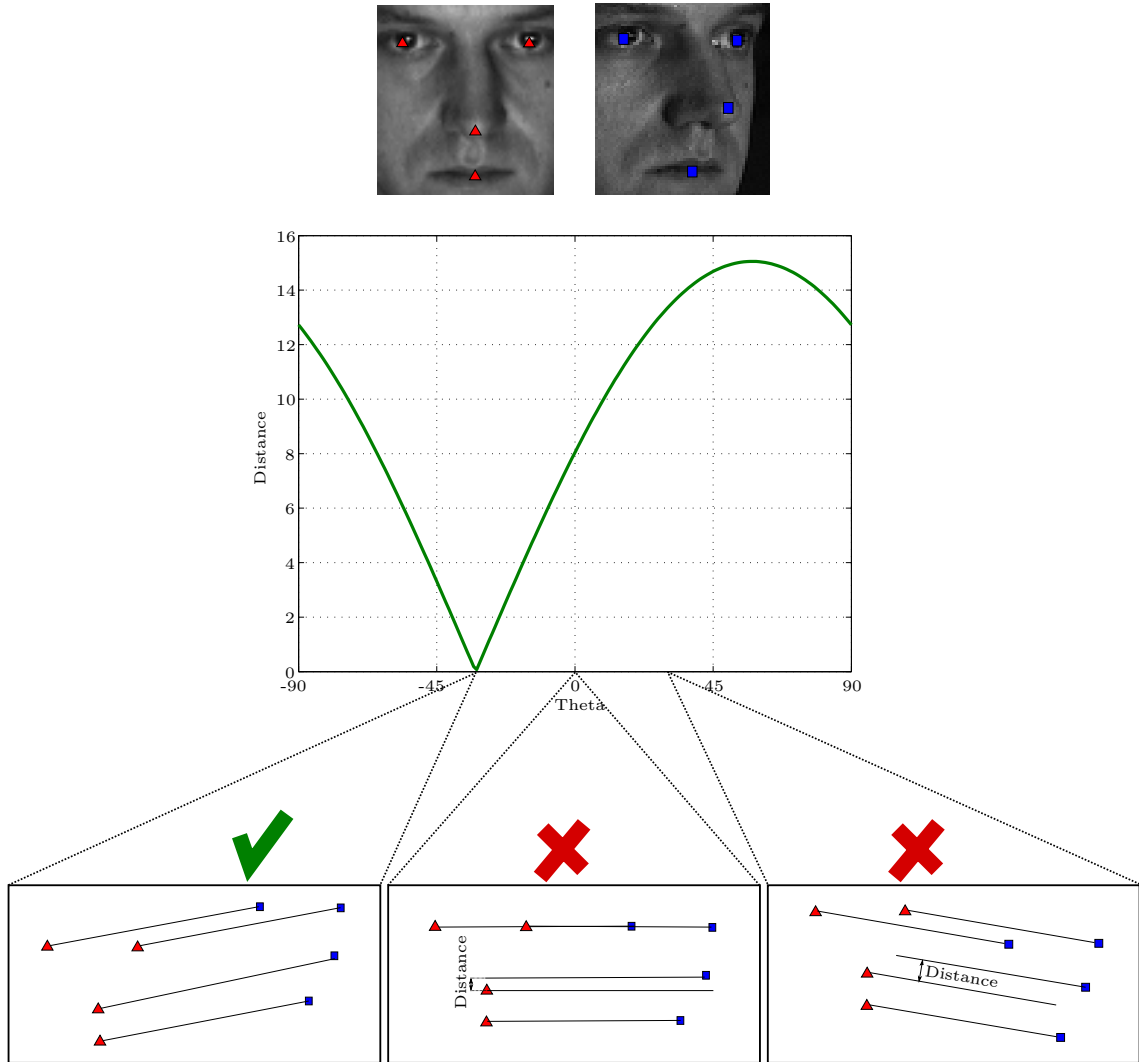


Figure 3.4: Example of our method to compute the epipolar under scaled orthographic projection. For each angle θ we compute the distance perpendicular to it of a fourth point in the two images. We choose the epipolar geometry that has the smallest distance.

(P_{2x}^i, P_{2y}^i) are the coordinates of the i 'th point in the second image, while $(l_{1x}^i, l_{1y}^i, l_{1c}^i)$ represents the line of slope $\tan(\theta)$ that passes through each of the points in the first image. Note that it is convenient to represent the lines in parametric form (in terms of $\sin(\theta)$ and $\cos(\theta)$) so that after multiplying the first two components of Eqn. 3.3 each restriction becomes:

$$(aP_{2x}^i + bP_{2y}^i + T_x, -bP_{2x}^i + aP_{2y}^i + T_y, 1) \begin{pmatrix} -\sin(\theta) \\ -\cos(\theta) \\ \sin(\theta)P_{1x}^i - \cos(\theta)P_{1y}^i \end{pmatrix} = 0 \quad (3.4)$$

The final constraint comes from the fact that T_x and T_y are not independent, they are constrained to be translations perpendicular to the angle of the epipolar lines in the first image θ :

$$\cos(\theta)T_x + \sin(\theta)T_y = 0 \quad (3.5)$$

We can think of $\sin(\theta)$ and $\cos(\theta)$ as separate variables, with the constraint $\sin(\theta)^2 + \cos(\theta)^2 = 1$. Then, with Eqns. (3.4) and (3.5), we have a system of bilinear and a quadratic equation. This has six unknowns, a , b , T_x , T_y , $\sin(\theta)$ and $\cos(\theta)$, and $n + 2$ equations, given n point correspondences. We solve this in a very simple way. Noting that the equations become linear when θ is known, we simply consider a brute-force sampling of θ , and check which value produces a consistent set of linear equations. For each θ we compute the alignment (a candidate epipolar geometry) given 3 points. When this has been done, we use the fourth point to compute, the quality of the alignment.

1. Use 3 points to solve for (a, b, T_x, T_y) using Eqns. (3.4) and (3.5).

2. Apply the similarity transform $M = \begin{pmatrix} a & b & T_x \\ -b & a & T_y \\ 0 & 0 & 1 \end{pmatrix}$ to the second image.
3. Use the distance of the 4th point in the direction perpendicular to θ to determine how good the match is. The best transformation M is the one that minimizes this distance.

The rectification procedure is, therefore, applying the best M to the second image and then rotating both images by θ in such a way that the epipolar lines become horizontal. After this is done, we are ready to compute the stereo matching cost to determine the image similarity.

3.2.2 Epipolar Geometry and Horizontal Movement

We will now study a particular case of the general setup: an upright person with both images taken with the camera located at the same height as the person's head (in fact, our reasoning applies to any situation in which the eyes and both camera focal points are coplanar). In that case we know that the epipolar lines are parallel to the lines connecting the eyes. For this case we only determine the epipolar geometry using three feature points. The two eyes will define the direction of the epipolar lines. This tells us θ . Given a correspondence between three points, Eqns. (3.4) and (3.5) then provide four linear constraints on four unknowns, allowing us to solve for the epipolar geometry linearly. Moreover, our experiments show that in many practical situations, even when the cameras are not perfectly at eye level these alignments work reasonably well.

Since this is the simplest alignment method we study, this procedure is, additionally, the base procedure we use to generate the thumbnails for the four-point alignment procedure explained in Section 3.2.1. The method presented in this section is equivalent to the case presented in Section 3.2.1 when $\theta = 0$.

We now describe a simple method of rectifying the two images so that horizontal rows of each image contain corresponding epipolar lines. Note that this rectification does not require that the three matched landmark points in the two images must coincide, just that corresponding points should lie on corresponding horizontal lines in the rectified images.

1. Rotate the image so the eyes are horizontal.
2. Scale the image so that the vertical distance between the eyes and the mouth is an arbitrary but fixed d .
3. Translate the images up/down in such a way that eyes are on an arbitrary but fixed line y_e .
4. Translate in the x direction so the center of mass of the x coordinates is 0. This step is not needed to align corresponding epipolar lines, but is convenient.
5. Cut a thumbnail in such a way that the height is arbitrary but fixed and the thumbnail includes the three feature points.

Note that this procedure will produce thumbnails that will have different widths but a fixed height. This is appropriate, since given our assumptions the apparent height of a face will be the same for all images, but its apparent width may vary with the viewing direction.

3.3 Stereo Matching and Face Recognition

There exist a wide variety of stereo algorithms. We require an efficient stereo algorithm appropriate for wide baseline matching of faces. We have used Criminisi et al. [22]¹ which has been developed for video conferencing applications and so seems to fit our needs. This algorithm handles slanted surfaces in an elegant yet limited way. It is not obvious that it will work for the large changes in viewpoint that can occur in face recognition, but we will show that it does.

In this section we will review the stereo matching method of Criminisi et al. [22] as it is presented by its authors. In the following section we will describe how we adapt the algorithm for the purpose at hand.

It is important to stress that we are relatively unaffected by some of the difficulties that make it hard to avoid artifacts in stereo reconstruction. For example, when many matches have similar costs, matching is ambiguous. One weakness of dynamic programming stereo algorithms is that when matching is ambiguous, it can be difficult to produce correspondences that are consistent across scan lines. Selecting the right match is difficult, but important for good reconstructions. Since we only use the cost of a matching, selecting the right matching is unimportant to us in this case. Also, errors in small regions, such as at occluding boundaries, can produce bad artifacts in reconstructions, but that is not a problem for our method as long as they don't affect the cost too much.

The core of the stereo method calculates a matching between two scanlines (rows of each face). The algorithm is a dynamic programming stereo matching algorithm that is fast and performs well when compared to other methods.

¹We also tried the method described in Cox et al. [21] and found the method to be about twice as fast but less accurate (about 8% on average on several gallery-probe experiments with a gallery of 68 individuals) than the method described in Criminisi et al. [22].

The algorithm accounts for exactly one pixel in one image with each step taken. Each step involves a transition from one point to another in four planes (or cost matrices) called C_{Lo} , C_{Lm} , C_{Ro} and C_{Rm} . Each point in a matrix represents the last point in each image that has been accounted for, along with the nature of the last step used to account for a point. Points are accounted for by matching (m) and occlusions (o) in the left (L) and right (R) images. The planes naturally define the persistence of states. By setting the state transition costs adequately many state transitions can be favored or biased against. For example long runs of occlusions can be favored over many short runs by setting a high cost for entering or leaving an occluded state. This formulation handles slanted surfaces well (because it allows many-to-one matches) and offers better control over the occlusion costs than traditional one plane models [21].

The elements of the cost matrix are initialized to $+\infty$ everywhere except in the right occluded plane where:

$$C_{Ro}[i, 0] = i\alpha \quad \forall i = 0 \dots W - 1 \quad (3.6)$$

α is the cost of a persistent occlusion.

The forward step of the 4-state DP computes the four cumulative cost matrices according to the following recurrence relation, in which β is the cost of beginning

an occlusion, and β' is the cost of ending one:

$$C_{Lo}[l, r] = \min \begin{cases} C_{Lo}[l, r - 1] + \alpha \\ C_{Lm}[l, r - 1] + \beta \\ C_{Rm}[l, r - 1] + \beta \end{cases} \quad (3.7)$$

$$C_{Lm}[l, r] = M(l, r) + \min \begin{cases} C_{Lo}[l, r - 1] + \beta' \\ C_{Lm}[l, r - 1] + \gamma \\ C_{Rm}[l, r - 1] \\ C_{Ro}[l, r - 1] + \beta' \end{cases} \quad (3.8)$$

where $M(l, r)$ is the cost of matching the l th pixel in the left scanline with the r th pixel in the right scanline. α , β , β' and γ are parameters that can be set experimentally. C_{Ro} and C_{Rm} are symmetric. Our experiments show that the method is rather insensitive to these parameters and all experiments shown here are run with $\alpha = 0.5$, $\beta = \beta' = 1.0$ and $\gamma = 0.10$ as recommended in [22]. $M(l, r)$ is a fast approximation to the normalized cross correlation of a 3×7 window around the points (l, s) and (r, s) of the images, where s is the current scanline.

The cost of matching the two scan lines l_1 and l_2 , denoted $\text{cost}(l_1, l_2)$, is: $C_{Ro}[l - 1, r - 1]$. The optimal matching solution will be a sequence of symbols in the alphabet: $\Sigma = \{C_{Lo}, C_{Lm}, C_{Ro}, C_{Rm}\}$ which can be obtained by following a backward step. A solution (a word in Σ^*) that encodes the optimal matching to a given matching problem between scanlines $I_{1,i}$ and $I_{2,i}$ has length equal to $|I_{1,i}| + |I_{2,i}|$. We have no use for the optimal matching itself, we only use its cost and its length to normalize it.

One of the key ingredients to the flexibility of this method is the ability to match multiple pixels in one scanline to one pixel in the other. This is done by concatenating several consecutive C_{Lm} (or C_{Rm}) in the word that encodes the solution.

3.3.1 Rectification and Matching Costs

When we match a probe image to different gallery images, we obtain different rectifications. While the original thumbnails are axial rectangles, the rectified thumbnails will be arbitrarily rotated rectangles that will contain varying numbers of rows with valid pixels, and different numbers of valid pixels in each row. It is therefore important to avoid any bias in our image comparisons which favor some thumbnail orientations over others. In this section we explain how to adapt Criminisi et al. [22] to match rectified images in which the length of scanlines varies.

The equations presented in Eqns. 3.7 and 3.8 are an effective measure of similarity when the two images are square and of identical size. When these assumptions are broken, Eqns. 3.7 and 3.8 stop being an effective measure of similarity because now in image comparisons there will be a different number of pixels in each image. We will focus this section in adapting this metric for the purpose at hand.

As previously mentioned, all solutions found by the method of Criminisi et al. have length equal to the sum of both scan lines being matched. This is due to the fact that the algorithm at every step accounts for exactly one pixel. However, since each cost is going to be compared to other costs matched over scanlines of potentially different lengths, we need some normalization strategy.

There are two sensible normalizations that can be used, one that weighs every row equally:

$$\text{cost}(I_1, I_2) = \frac{1}{n} \sum_{i=1}^n \frac{\text{cost}(I_{1,i}, I_{2,i})}{|I_{1,i}| + |I_{2,i}|} \quad (3.9)$$

and one that weighs every match (arc in the graph) equally. This is the one we actually use:

$$\text{cost}(I_1, I_2) = \frac{\sum_{i=1}^n \text{cost}(I_{1,i}, I_{2,i})}{\sum_{i=1}^n |I_{1,i}| + |I_{2,i}|} \quad (3.10)$$

The cost expressed in Eqn. 3.10 is a sensible measure of similarity since it is not dependent on the relative scale of the images, it just calculates the average cost per match made (that is per arc in the graph) over all scan lines. However, the costs in Eqns. 3.9 and 3.10 are built on top of the structure of the match found. This property is useful because it makes the cost not depend on the shape of the non-data that is present in the image, and therefore there will be no biases towards matches with scan lines in both images at the same angles.

We identify two special cases to Eqns. (3.9) and (3.10):

1. When two scan lines of non-data are being matched
2. When a scan line of non-data is being matched to scan line of data

We could pay a constant penalty for each of these special situations but doing so would artificially add noise to the similarity cost. We decide to not include these special cases in the average described in Eqn. (3.10) and let the other pixels, for which there is actual data to match, decide what the average cost per match should be.

Let $\text{cost}(I_1, I_2)$ be defined as either of the two cases studied above. Since we do not know which image is left and which image is right we have to try both options. One of them will be the true cost, the other cost will be noise and should be ignored.

$$\text{similarity}(I_1, I_2) = \min \begin{cases} \text{cost}(\text{rectify}(I_1, I_2)) \\ \text{cost}(\text{rectify}(I_2, I_1)) \\ \text{cost}(\text{rectify}(\text{flip}(I_1), I_2)) \\ \text{cost}(\text{rectify}(I_2, \text{flip}(I_1))) \end{cases} \quad (3.11)$$

Additionally, *flip* produces a left-right reflection of the image and adjusts the hand clicked positions of the four points accordingly. *flip* is helpful when two views see mainly different sides of the face. In this case, a truly correct correspondence would mark most of the face as occluded. However, since faces are approximately vertically symmetric, *flip* approximates a rotation about the y axis that creates a virtual view so that the same side of the face is visible in both images. For example, if we viewed a face in left and right profile, there would be no points on the face visible in both images, but flipping one image would still allow us to produce a good match. *rectify* performs the rectification described in the 4-point case, or in the 3-point case does nothing at all, since all images are already partially rectified to handle this case.

Finally, we perform recognition simply by matching a probe image to the most similar image in the gallery. For the method to work well all the images in the gallery should be in the same pose.

Before closing this section it is important to note how simple the proposed approach is. It is a two step process: (1) alignment according to assumptions

regarding the viewing conditions, (2) similarity computation using stereo matching. In the next section we will see that this very straight-forward approach demonstrates excellent performance.

3.4 Experiments

We have tested our algorithm using the CMU PIE database [70]. This database consists of 13 poses of which 9 have approximately the same camera altitude (poses: c34, c14, c11, c29, c27, c05, c37, c25 and c22). Three other poses that have a significantly higher camera altitude (poses: c31, c09 and c02) and one last pose that has a significantly lower camera altitude (pose c07). We say that two poses have aligned epipolar lines if they are both from the set: {c34, c14, c11, c29, c27, c05, c37, c25, c22}. If not, we say that two poses have misaligned epipolar lines.

The thumbnails used were generated as described in Section 3.2.2. All images have a height of 72, a pose-dependent width and a distance between the eyes and the mouth of $d = 50$ and the eyes are horizontally located in $y_e = 13$. For the 3-point Stereo Matching Distance (3ptSMD) this is all the image processing performed, the stereo matching cost was then computed and normalized and this cost is the image similarity between the two faces. For the 4-point Stereo Matching Distance (4ptSMD) the epipolar rectification was then performed on the thumbnail. After rectification, the stereo matching cost was computed and this cost is the image similarity between the two faces.

A number of prior experiments have been done with pose variation using the CMU PIE database, but somewhat different experimental conditions. We have run our own algorithm under a variety of conditions so that we may compare to these.

For example, to compare results with [33, 35, 17] we need to use a subset of 34 people because they use 34 people for training and the remaining 34 for testing. We do not require training, but we are interested in comparing the methods in equal conditions so we tested on individuals 35-68 from the PIE database. To compare with [63] we used 68 people as a test set. Then to illustrate that our method works in more realistic situations we evaluated simultaneous variation in pose and illumination. This too is done in two separate experiments, one to compare with [33, 35] and one to compare with [63].

3.4.1 PIE Pose Variation: 34 Faces

We conducted an experiment to compare our method with four others. We compared with two variants of eigen light-fields[33], eigenfaces[77] and FaceIt as described in [33, 35]. FaceIt² is a commercial face recognition system from Identix which finished top overall in the Face Recognition Vendor Test 2000. Eigenfaces is a common benchmark algorithm for face recognition. Finally, eigen light-fields is a state of the art method for face recognition across pose.

In this experiment we selected each gallery pose as one of the 13 PIE poses and the probe pose as one of the remaining 12 poses, for a total of 156 gallery-probe pairs. We evaluated the accuracy of our method in this setting and compared to the results in [33, 35]. Table 3.3 summarizes the average recognition rates. Table 3.1 presents detailed results for this experiment using 3ptSMD and Table 3.2 presents detailed results for this experiment using 4ptSMD. Figure 3.5 shows several cross-sections of the results with different fixed gallery poses.

²Version 2.5.0.17 of the FaceIt recognition engine was used.

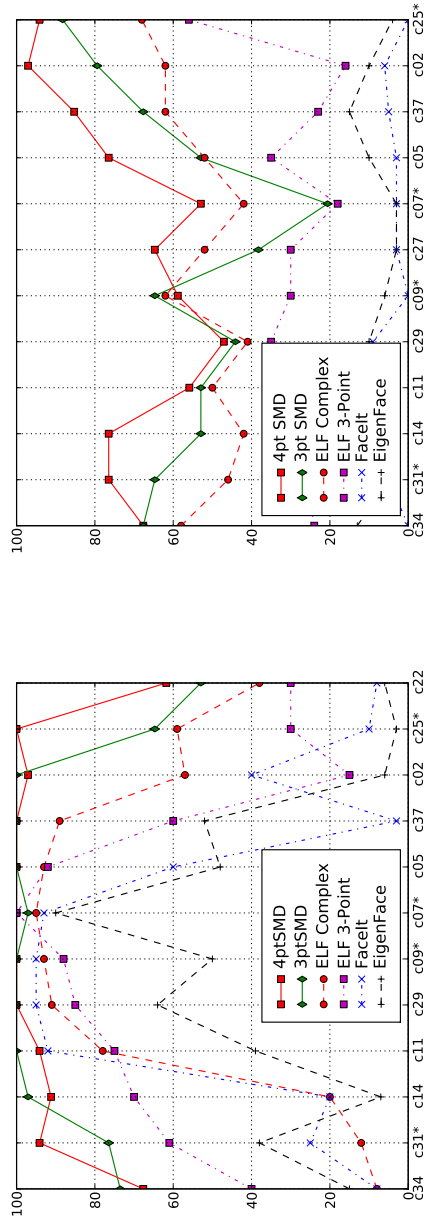
The fact that 3ptSMD performs solidly both when the epipolar lines fit (with an average of 81.4%) and when they don't (with an average of 75.4%) and overall (with an average of 78.5% as reported in Table 3.7) shows that assuming horizontal epipolar geometry is not a bad approximation for real applications of face recognition across pose, even when this assumption does not hold perfectly.

Table 3.1: Results for pose variation for 34 faces with 3ptSMD. The diagonals are not included in any average. The table layout is the same as [63] and [36]. The global average for this table is 79.8%.

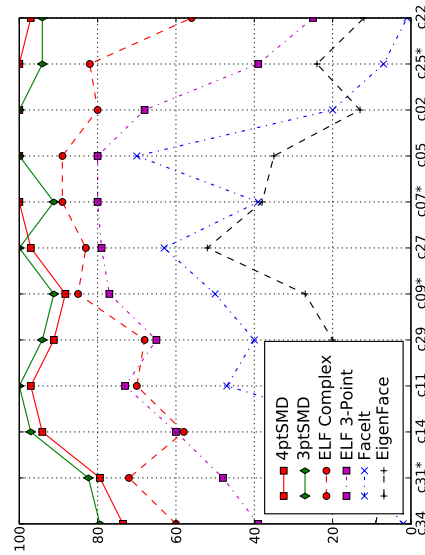
azimuth altitude Probe Pose	-66 3 c34	-47 13 c31	-46 2 c14	-32 2 c11	-17 2 c29	0 15 c09	0 2 c27	0 1.9 c07	16 2 c05	31 2 c37	44 2 c25	44 13 c02	62 3 c22	avg
Gallery Pose														
c34	-	91	82	74	62	32	35	18	50	56	65	71	71	58
c31	94	-	88	88	76	91	59	32	56	65	97	82	76	75
c14	94	91	-	100	100	82	91	76	88	82	56	88	56	83
c11	94	97	100	-	100	88	100	94	94	97	53	94	65	89
c29	88	88	100	100	-	100	100	100	100	97	62	94	53	90
c09	59	100	76	94	100	-	97	82	97	88	97	82	79	87
c27	74	76	97	100	100	100	-	97	100	100	65	100	53	88
c07	29	41	74	91	97	79	100	-	97	82	38	65	24	68
c05	68	76	100	97	100	97	100	94	-	100	85	100	79	91
c37	79	82	97	100	94	91	100	91	100	-	94	100	94	93
c25	47	94	44	59	44	91	47	18	68	85	-	79	97	64
c02	79	79	94	91	88	82	94	62	100	97	94	-	94	87
c22	68	65	53	53	44	65	38	21	53	68	88	79	-	57

Table 3.2: Results for pose variation for 34 faces with 4ptSMD. The diagonals are not included in any average. The table layout is the same as [63] and [36]. The global average is 86.82%

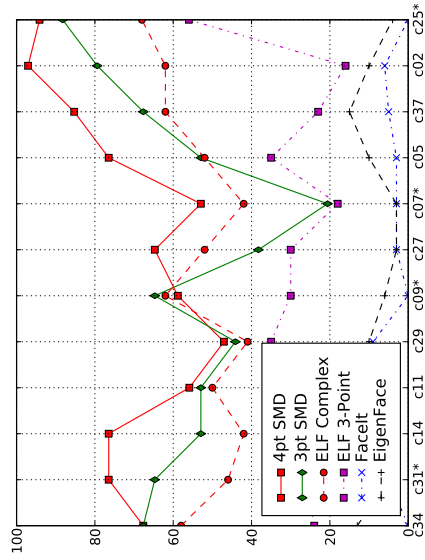
azimuth altitude Probe Pose	-66 3 c34	-47 13 c31	-46 2 c14	-32 2 c11	-17 2 c29	0 15 c09	0 2 c27	0 1.9 c07	16 2 c05	31 2 c37	44 2 c25	44 13 c02	62 3 c22	avg
Gallery Pose														
c34	-	91	91	82	68	44	44	35	50	53	65	74	65	63
c31	97	-	100	97	97	94	76	56	65	74	91	82	76	83
c14	100	100	-	100	97	85	91	71	91	82	68	91	85	88
c11	97	97	100	-	100	94	94	97	100	97	82	94	74	93
c29	85	97	97	100	-	100	97	100	97	97	85	94	53	91
c09	62	97	91	100	100	-	100	97	100	97	91	91	76	91
c27	68	94	91	94	100	100	-	100	100	100	100	97	62	92
c07	41	71	79	97	100	100	100	-	100	97	85	94	35	83
c05	79	85	100	100	97	94	100	100	-	100	100	100	91	95
c37	74	79	94	97	91	88	97	100	100	-	100	100	97	93
c25	76	88	68	76	82	88	97	82	100	100	-	97	97	87
c02	88	85	88	85	85	94	91	94	97	100	97	-	100	92
c22	68	76	76	56	47	59	65	53	76	85	94	97	-	71



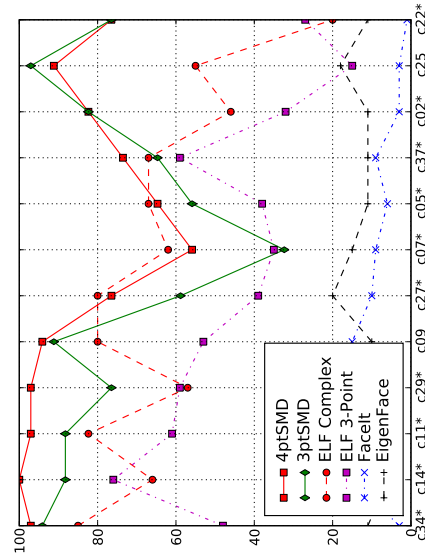
(a) Gallery Pose c27



(c) Gallery Pose c37



(b) Gallery Pose c22



(d) Gallery Pose c31

Figure 3.5: Cross-sections with fixed gallery pose for the results presented in Table 3.3. Probe poses marked with * have a vertical misalignment of about 10 degrees with the corresponding gallery pose.

Figure 3.5 shows a comparison with the results presented in the paper of Gross et al. [33, 35]. In this experiment we observe that in all gallery poses our method outperforms all the other methods for the extreme probe poses (c34, c31, c14, c02, c25 and c22). Observe that the 4ptSMD method is considerably better than than 3ptSMD at the poses where there is considerable misalignment (the poses marked with *).

Table 3.4 shows a comparison with Chai et al. [17], using the experimental conditions described in their paper. The gallery pose is c27 and contains 34 faces, the probe poses are: c05, c29, c37, c11, c07 and c09. Note that this is a slice of data from Table 3.1 . Our 3ptSMD method produces nearly perfect results in these conditions, results that are much better than those reported in Chai et al.

Table 3.3: A comparison of our stereo matching distance with other methods across pose.

34 Faces	
Method	Accuracy
Eigenfaces [33, 35]	16.6%
FaceIt [33, 35]	24.3%
Eigen light-fields (3-point norm.) [33, 35]	52.5%
Eigen light-fields (Multi-point norm.) [33, 35]	66.3%
3-point Stereo Matching Distance	79.8%
4-point Stereo Matching Distance	86.8%

68 Faces	
Method	Accuracy
LiST (Romdhani et al. [63])	74.3%
3-point Stereo Matching Distance	74.5%
4-point Stereo Matching Distance	82.4%

Table 3.4: Comparisons over a slice of the data with the method of Chai et al. [17] and Gross et al. [35]. The gallery pose is c27 and contains 34 faces. The table layout is the same as the one presented in [17].

Probe Pose	Methods			
	3ptSMD	LLR-step5 with PCA+LDA	ELF (3-P Normalization)	ELF (Complex)
c05	100%	98.5%	88%	93%
c29	100%	100%	86%	91%
c37	100%	82.4%	74%	89%
c11	97%	89.7%	76%	78%
c07	100%	98.5%	100%	95%
c09	100%	98.5%	87%	93%
Mean	99.5%	94%	85.1%	89.8%

3.4.2 PIE Pose Variation: 68 Faces

We also compared our results with the ones presented in Romdhani et al. [63]. These results are, to our knowledge, the best reported on the whole PIE database for pose variation. In this work all 68 images were used, so for this part we report our results using all 68 faces. Table 3.3 summarizes the results of this experiment.

The global average for the method of Romdhani et al. [63] is 74.3%, the global average for our 3ptSMD method is about the same, at 74.5%. For the subset of poses in which the epipolar lines fit perfectly our average performance is 80.8%, while theirs is 71.6%. We consider the case where all epipolar lines fit to be the best possible scenario for the 3ptSMD. When the epipolar lines are misaligned the average for 3ptSMD is 69.2%. Our 4ptSMD achieves overall accuracy of 82.4%, which is considerably higher than the performance of Romdhani et al. Our method runs about 40 times faster than the method presented in [63], requires fewer manually specified points, and is much simpler. Detailed results are presented in Tables 3.5 and 3.6.

Table 3.5: Confusion matrix for pose variation for 68 faces with 3ptSMD. The diagonals are not included in any average. The table layout is the same as [63] and [36]. The global average for this table is 74.5%.

azimuth altitude Probe Pose	-66 3 c34	-47 13 c31	-46 2 c14	-32 2 c11	-17 2 c29	0 15 c09	0 2 c27	0 1.9 c07	16 2 c05	31 2 c37	44 2 c25	44 13 c02	62 3 c22	avg
Gallery Pose														
c34	-	79	85	74	59	29	37	15	47	51	49	60	54	53
c31	84	-	81	68	60	78	44	22	43	54	90	68	65	62
c14	96	85	-	100	100	76	93	60	79	82	56	82	50	80
c11	94	90	100	-	100	88	100	90	90	96	51	90	53	86
c29	88	79	100	100	-	99	100	97	100	96	54	90	50	87
c09	44	96	72	88	97	-	97	76	96	91	93	79	66	82
c27	60	62	93	100	100	100	-	97	100	100	62	97	46	84
c07	25	34	72	87	96	76	97	-	97	85	31	62	16	64
c05	60	60	90	91	100	97	100	96	-	100	76	100	63	86
c37	74	69	93	97	94	91	100	84	99	-	88	100	79	88
c25	44	93	35	40	41	85	40	16	66	79	-	72	88	58
c02	75	74	87	88	76	68	94	60	96	99	85	-	88	82
c22	56	62	47	43	37	53	32	13	41	56	87	69	-	49

Table 3.6: Confusion matrix for pose variation for 68 faces with 4ptSMD. The diagonals are not included in any average. The table layout is the same as [63] and [36]. The global average for this table is 82.4%

azimuth altitude Probe Pose	-66 3 c34	-47 13 c31	-46 2 c14	-32 2 c11	-17 2 c29	0 15 c09	0 2 c27	0 1.9 c07	16 2 c05	31 2 c37	44 2 c25	44 13 c02	62 3 c22	avg
Gallery Pose														
c34	-	79	91	78	65	38	44	26	50	50	60	71	56	59
c31	91	-	99	96	94	78	65	50	62	65	84	72	60	76
c14	97	100	-	97	91	87	79	71	79	76	59	76	78	82
c11	94	97	99	-	100	97	94	94	88	94	79	87	65	90
c29	87	97	96	100	-	100	99	100	96	94	82	81	53	90
c09	54	91	84	99	100	-	100	97	94	94	85	90	65	87
c27	60	93	91	97	99	99	-	100	97	99	97	97	62	90
c07	40	62	79	97	100	96	100	-	100	99	88	97	32	82
c05	71	79	90	93	97	97	99	100	-	100	100	99	78	91
c37	66	74	85	94	90	91	97	99	100	-	100	100	91	90
c25	65	79	56	66	71	85	91	79	97	100	-	99	94	81
c02	81	71	74	81	69	93	90	85	93	100	99	-	99	86
c22	57	62	66	56	44	49	47	35	66	76	88	91	-	61

3.4.3 PIE Pose and Illumination Variation

We also evaluated the performance of the method across pose and illumination. Although our method is not designed to handle lighting variation, the use of normalized correlation in matching may provide some robustness to lighting changes. The objective of this experiment is to verify that the good performance obtained when there is variation in pose (the previous experiments) are not an artifact of the (constant) illumination condition, and that the system degrades gracefully with lighting changes.

In this section we compare our method to Bayesian Face Subregions (BFS) [33] in the case of simultaneous variation of pose and illumination. For this experiment, the gallery is frontal pose and illumination. For each probe pose, the accuracy is determined by averaging the results for all 21 different illumination conditions. The results of this comparison are presented in Figure 3.6. We observe that our algorithm strictly dominates BFS over all probe poses.

For lighting invariance they use [34] which computes the reflectance and illumination fields from real images using some simplifications, while we simply use an approximation to normalized correlation.

Table 3.7: Summary of the cases where the camera movement is horizontal and when it is not over the experiments with 3ptSMD and 4ptSMD.

Method	# Faces	Epipolar Alignment	Epipolar Misalignment	Average
3ptSMD	34	84.8%	75.6%	79.8%
3ptSMD	68	80.8%	69.2%	74.5%
4ptSMD	34	87.2%	86.5%	86.8%
4ptSMD	68	82.6%	82.3%	82.4%

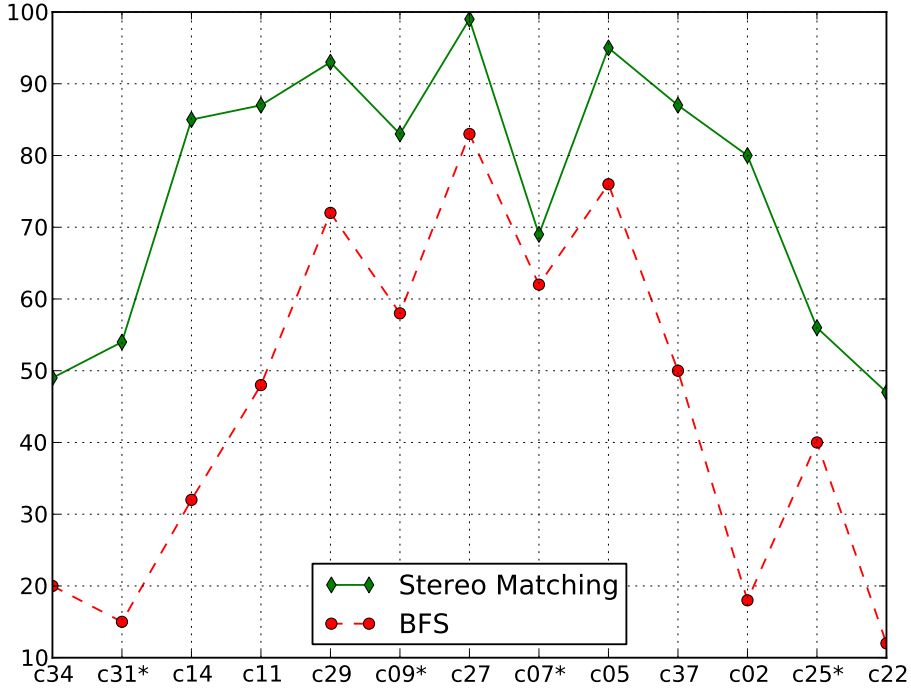


Figure 3.6: (A comparison of our method with BFS. Gallery pose is frontal (c27) probe poses are as indicated in the x axis, we report the average over the 21 illuminations).

We also performed experiments in such a way that we can compare with [9] and [63]. For this experiment we used images of the faces of 68 individuals viewed from 3 poses (front: c27, side: c5 and profile: c22) and illuminated from 21 different directions. We used light number 12 for the gallery illumination to be able to compare our results with [63]. They select that lighting because “...the fitting is generally fair at that condition”. Our results are presented in Table 3.8. We do not expect our results to be as good as those of [63], because our algorithm only accounts for lighting variation by using a fast approximation to normalized cross correlation as described in Criminisi et al. [22], while [63] has a 3-D model and performs an optimization to solve for the lighting that best matches the model to

the image. We also tested on without ambient lights part of the PIE dataset which has harsh shadows. Other works have not reported results without ambient lights.

Table 3.8: Accuracy percentage with pose and illumination variation. The cell format is: (with ambient lights)/(without ambient lights). Three galleries and three probes were used. F: Frontal, S: Side, P: Profile. The table layout is similar to [63]

light	F Gallery			S Gallery			P Gallery		
	F	S	P	F	S	P	F	S	P
	2	94/38	93/44	32/4	85/41	100/53	41/6	26/21	18/25
3	96/68	96/76	35/13	93/65	100/85	41/9	29/25	16/25	100/51
4	97/82	94/87	37/24	96/82	100/94	35/12	34/25	24/31	100/66
5	99/100	99/97	35/34	99/97	100/100	47/32	38/35	25/29	100/94
6	100/99	100/99	41/35	100/99	100/100	57/56	38/29	43/24	100/100
7	100/99	99/97	37/34	99/87	100/100	53/49	29/21	35/16	100/100
8	100/100	100/100	44/37	100/100	100/100	56/60	35/19	43/25	100/100
9	100/100	100/100	44/44	100/100	100/100	65/62	40/35	47/46	100/100
10	99/90	99/93	29/34	99/88	100/99	49/35	32/28	28/21	100/87
11	100/100	100/100	46/44	100/100	100/100	60/56	47/32	49/35	100/100
12	-/-	100/100	53/44	100/100	-/-	71/62	49/46	56/53	-/-
13	100/100	100/100	46/41	100/100	100/100	63/49	44/43	49/49	100/100
14	100/100	100/100	47/43	100/100	100/100	66/49	44/46	59/53	100/100
15	100/100	99/94	46/31	100/100	100/100	54/40	37/46	60/54	100/100
16	100/100	97/74	40/21	100/97	100/99	51/32	40/41	53/47	100/100
17	100/90	96/49	35/19	99/75	100/84	49/26	32/41	44/47	100/100
18	99/91	99/97	37/28	99/90	100/97	38/25	35/37	22/32	100/79
19	100/100	100/99	38/29	100/99	100/100	54/38	43/35	44/32	100/99
20	100/100	100/100	44/38	100/100	100/100	63/51	49/41	51/40	100/100
21	100/100	100/100	50/40	100/100	100/100	65/54	47/47	57/53	100/100
22	100/100	100/99	50/37	100/100	100/100	57/40	38/46	60/54	100/100
avg	99/92	98/90	41/32	98/91	100/95	54/40	38/35	42/37	100/91

Our stereo matching method degenerates into an approximation to normalized correlation over small windows when there is no change in pose. Our method performs better than Romdhani et al. [63] when there is no pose change (gallery probe combinations: F-F, S-S and P-P). It is surprising that our method works better than theirs in this case because we are using a simple illumination insensitive image comparison technique and they perform an optimization to solve for lighting. Overall, for this experiment our global average is 74.6% while the global average of Romdhani et al. [63] is 81%, which is considerably better.

3.5 Conclusion

We have presented a simple, general method for face recognition with pose variation that is based on stereo matching. Our approach is motivated by the observation that correspondence is critical for face recognition across pose. Finding correspondences in 2-D is exactly the problem that stereo matching solves. We use stereo matching for face recognition across pose and show that this method exhibits excellent performance when compared to existing methods.

Our method is very simple. The formulation itself is straight-forward yet it is based on a very well-understood problem (stereo matching). The implementation can be done in C in a couple hundred lines of code.

The method we presented also degrades gracefully in the case of simultaneous variation of pose and illumination. Although our method is not really meant to handle lighting variation, since it uses normalized correlation it is somewhat robust to changes in illumination.

We evaluated our method using the CMU PIE dataset under a wide variety of conditions. Our results show that with pose variation and constant illumination our method is much more accurate than the methods of Gross, et al. [35], Chai et al. [17] and Romdhani et al. [63]. Additionally, our method is robust to some variation in lighting.

We feel that the main difference between our method and prior approaches is the use of stereo matching to find correspondences. Our method compares corresponding pixels very simply, using normalized correlation; this is a much more naive comparison than in many prior approaches. Therefore, we feel that the main reason for the superior experimental performance of our system lies in our emphasis on comparing images based on these correspondences.

Chapter 4

Face Recognition with Weight Variation

This chapter presents our work using stereo matching as method to recognize face in the presence of weight variation, this part of the work emphasizes the importance of finding correspondences and shows the feasibility of using stereo, even when the deformation is not rigid (such as weight variation and slight variation in expression).

There have been a wide variety of algorithms proposed for face recognition in the presence of important variations such as: pose, illumination, expression and aging, but to our knowledge no work has been done on weight variation. Our work evaluates some of these algorithms using our weight variation dataset. We believe that the availability of a weight variation dataset will encourage the development of algorithms that specifically account for weight variation.

4.1 Face Recognition with Weight Variation

We are also interested in considering another such type of face variation, changes in weight. Many applications, such as passport photo verification, police investigations, or the sorting of personal photographs require that we recognize an

individual in photos taken months or years apart, in which the subject’s weight may change considerably. Yet there has been no study of the effect that weight change has on the accuracy of recognition algorithms. We are interested in addressing this problem for the first time.

We have collected our own dataset of images with weight variation. In order to minimize the amount of time between photos. Some images were obtained from weight loss forums and personal photo documentaries. Other images came from the TV show *The Biggest Loser*. Figure 4.1 shows an example of an individual’s weight variation.

We performed some preliminary experiments to get a sense of the data set. In our experiments, we have found that performance of existing algorithms degrades markedly as the amount of weight change increases. This suggests that weight change alone can have a large effect on recognition performance.

Third, we find that there are large differences in the relative performance of different algorithms as the amount of weight change varies. In particular, we find that of the recognition algorithms tested, the most robust performance is obtained by algorithms that stress finding correspondences.

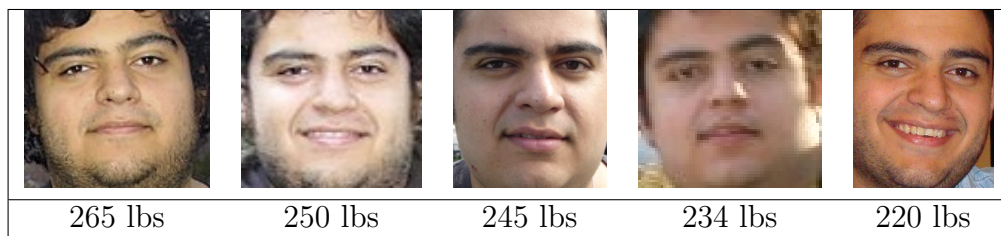


Figure 4.1: Facial changes as weight variation increases (images shown with permission of subject).

4.1.1 Experimental Evaluation

We used the following algorithms:

- **NC:** Normalized correlation.
- **Window-NSSD:** Window NSSD with clipping.
- **FPLBP:** Four patch local binary pattern as described in Wolf et al. [82] using code provided by its authors and default parameters and SSD of the descriptors of the two images. The images are filtered using a non-linear noise removal method (`wiener2` in MATLAB).
- **SMD-0d:** Stereo Matching Distance at zero-disparity, to evaluate the benefit of having a structured occlusion cost but no non-trivial correspondences.
- **SMD:** Stereo Matching Distance by Castillo and Jacobs [13].
- **SVM-diff:** SVM trained on “differences” of face images normalized to zero mean and unit variance [60]. The γ and C parameter are evaluated by 5-fold cross validation on the training set on a grid of options for (γ, C) [18].
- **SVM-GO:** SVM trained on gradient orientation “differences” [46]. The γ and C parameter are evaluated by 5-fold cross validation on the training set on a grid of options for (γ, C) [18].
- **LBP-SVM:** An SVM is trained to integrate several LBP-based distance measures from FPLBP and TPLBP. The LBP descriptors are computed using code publicly available from Wolf et al. [82]. The images are filtered using a non-linear noise removal method (`wiener2` in MATLAB).
- **ERCF:** The images are classified using ERCF. The costs are computed using the Linux binaries publicly available from Nowak and Jurie [54].

Figure 4.2 shows how the performance of non-learning algorithms varies with weight change. First, we can see that performance of all algorithms drops as the amount of weight change increases. The magnitude of these changes suggest that weight change plays a very significant role in the difficulty of this task.

We can also see that different algorithms display different levels of robustness to weight change. SMD is best when there is larger weight variation, but not when the weight change is small. FPLBP descriptors work very well when there is little weight variation but the performance decreases dramatically even in the presence of moderate weight gain.

Figure 4.2 also shows that there is a very slight difference between the performance of the two occlusion methods (window-nssd and SMD-0d). The performance of SMD-0d is slightly more robust to weight variation than window-nssd.

Figure 4.3 shows an ROC curve of all the non-learning methods compared on the entire Web Forum Dataset. This figure shows that SMD clearly and uniformly performs best.

From Figures 4.2 and 4.3 we observe that the two occlusion methods (window-nssd and SMD-0d) perform essentially equally well suggesting that it is not the treatment of occlusions but the ability to form correspondences with non-zero disparity that explains the difference between these two methods and SMD.

Figure 4.5 shows an ROC curve of all the methods that use learning. For this experiment we use half the dataset to train and half the dataset to test in a 2-fold cross-validation experiment. The curves presented are averages of each leg of the experiment. SMD (which was the best performing of the image matching methods) was also evaluated on the same testing set.

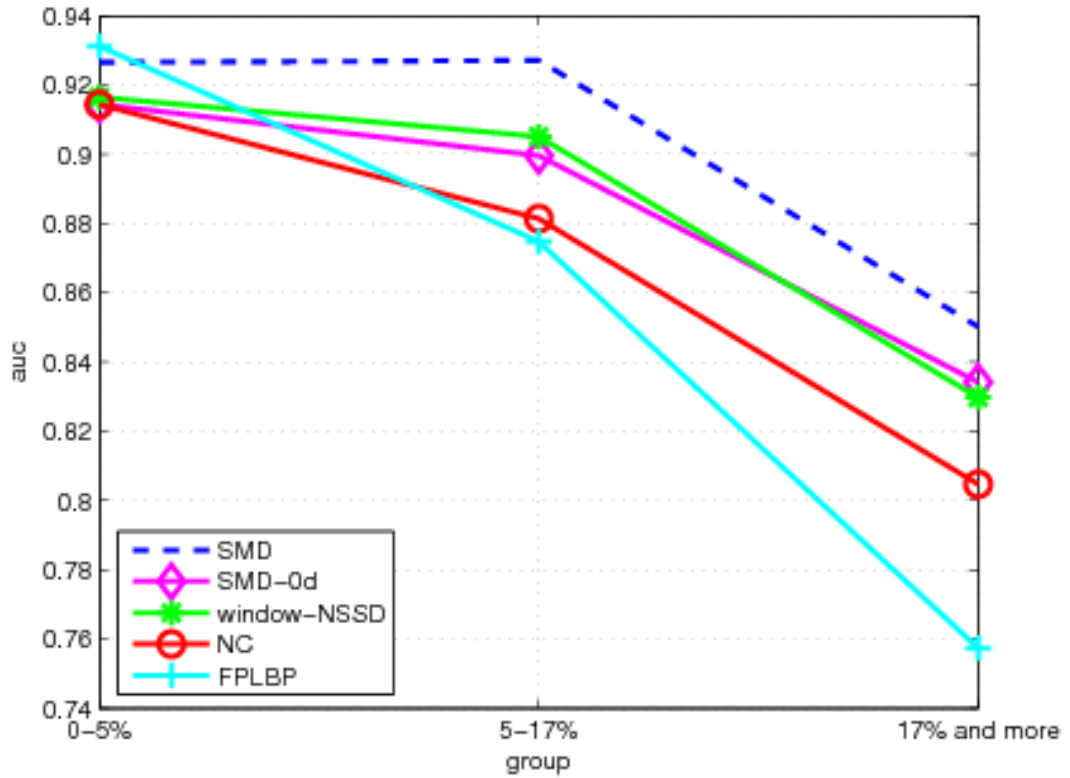


Figure 4.2: Performance of classifiers by groups of similar relative weight variation.

From Figure 4.5, we observe that ERCF and LBP-SVM perform best among the methods based on learning. The performance of SMD (which is not a learning based method) is better than the performance of the top two learning methods.

From Figures 4.4 and 4.5 we observe that the performance of LBP-SVM is globally very good but note that of all the evaluated methods the performance of LBP-SVM degrades the most as weight change increases, therefore, the method is the least robust to weight variation of all the learning methods evaluated.

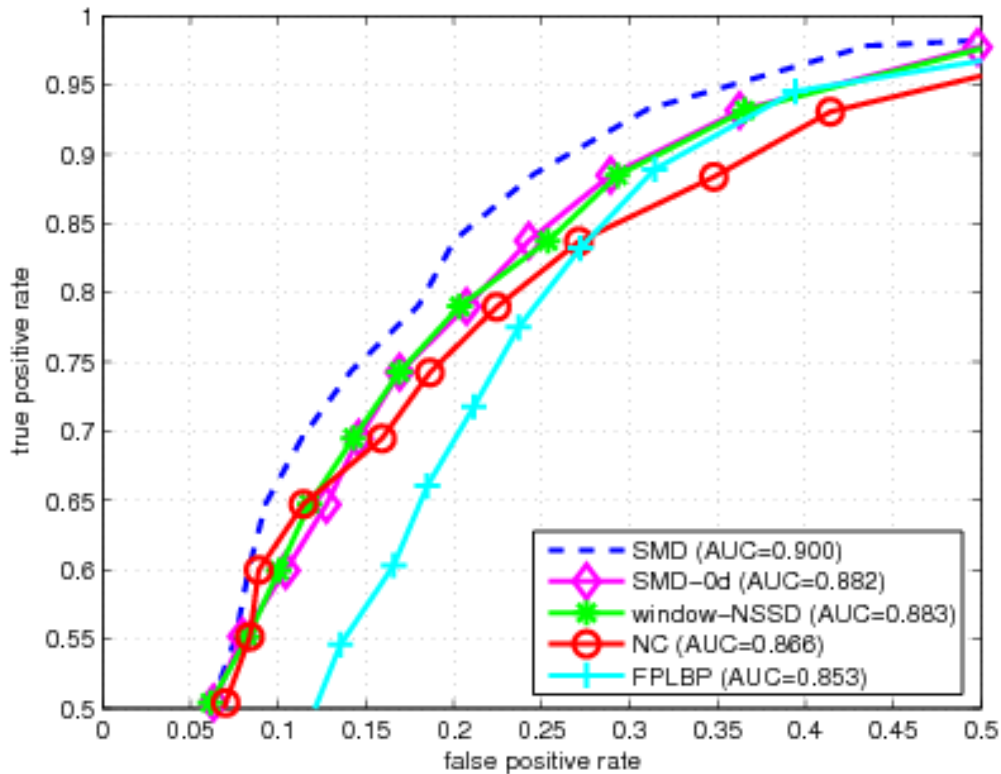


Figure 4.3: ROC curve comparing all the non-learning based methods.

4.1.2 Discussion

First, results with all algorithms show that weight change can have a very significant effect on the accuracy of recognition algorithms. We consider group one to contain minor fluctuations in weight, zero to eight pounds for a 160 pound person. Group two contains weight changes that are commonly seen over a few years time, ranging from eight to twenty-seven pounds for a 160 pound person. Group three contains more extreme weight changes. Depending on the algorithm, the moderate weight changes in group two can account for an increase of between 10% and 50% in errors from group one to group two. The more extreme weight changes of group three create much more dramatic increases in error rates. This indicates that our

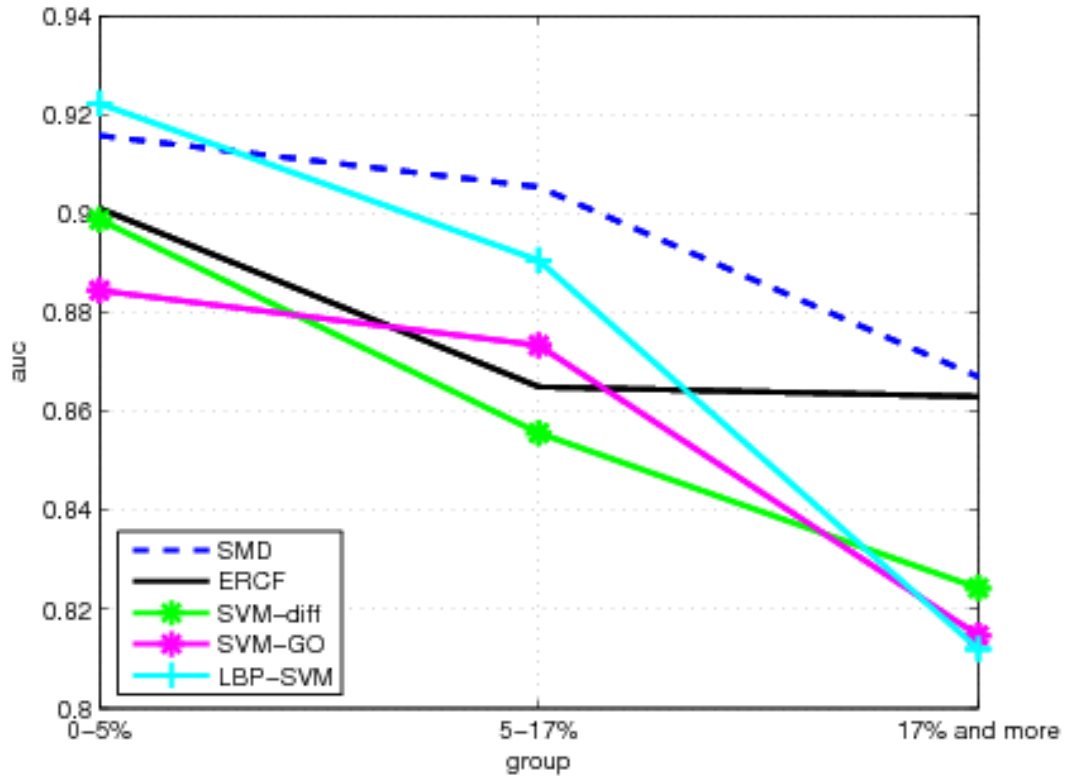


Figure 4.4: Performance of learning-based classifiers by groups of similar relative weight variation.

dataset does indeed capture many of the special difficulties posed by weight change, and that weight change is an important challenge for face recognition algorithms.

Next we will discuss why some methods work better than others in the presence of weight variation. The top two methods (ERCF [54] and SMD) have the common feature of finding non-trivial correspondences beyond those provided by alignment with a similarity transformation. While both methods do so taking very different approaches, experiments suggest that this results in better performance. These two methods do not explicitly account for weight variation but perform better in the presence of weight change, and are more robust to weight variation than other methods. For instance, methods based on local binary patterns (LBPs) perform

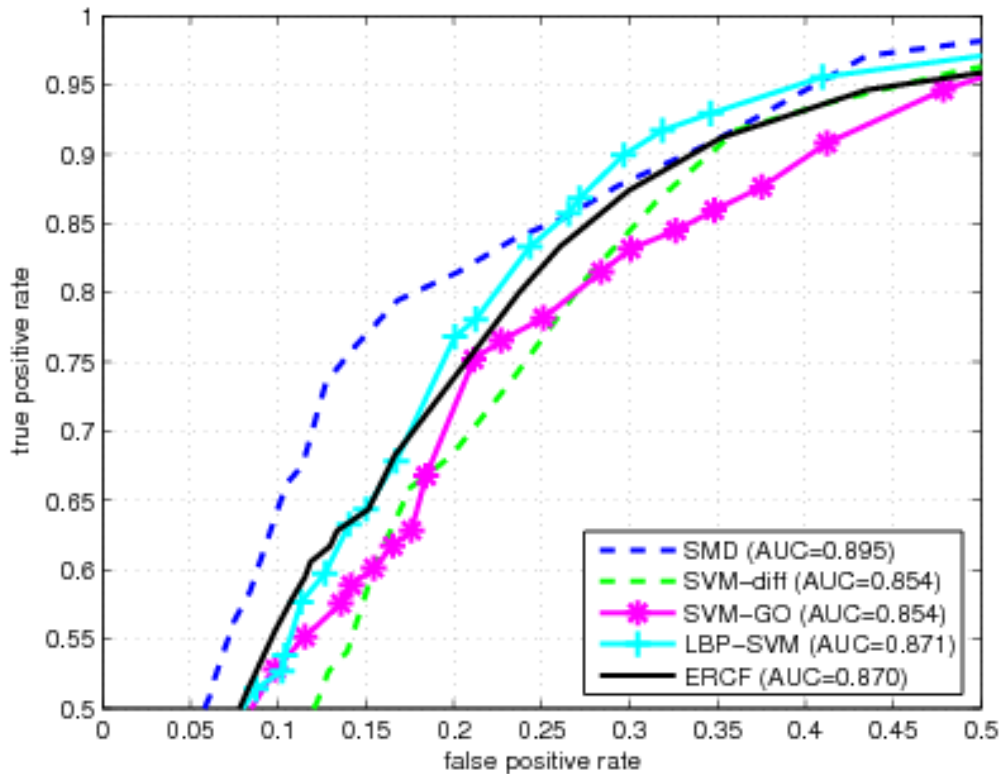


Figure 4.5: ROC curve comparing all the evaluated learning based methods. SMD was evaluated on the same testing set for comparison purposes.

remarkably well when there is little or no weight variation, but performance degrades rapidly when there is a large amount of weight variation.

The importance of correspondences is highlighted by the fact that SMD-0d is identical to SMD except that it only allows zero disparities. Therefore, the difference in performance between these two methods shows explicitly the importance of non-zero disparity correspondences.

Additionally the matching experiments show small yet significant difference in performance between normalized correlation and window-nssd with clipping and SMD-0d (the later two performing basically equally well). This illustrates the gain

in accounting for occlusions. It is, however, unclear how such knowledge can be leveraged in a learning based method.

As there is limited data it is hard to draw definitive conclusions about learning algorithms. But we have verified that learning-based methods can perform well with this data. It is somewhat difficult to determine what constitutes a realistic scenario for learning algorithms when there is weight change. On one hand, it may be possible in the future to train learning systems with more data. On the other hand, our data mainly consists of image pairs with weight change. In many situations, learning based methods will be trained mostly using pairs that have limited weight change, which might hinder their ability to account for weight changes that do occur.

We have evaluated a variety of existing methods on our dataset. We have shown that weight variation is a significant confounding factor in face recognition and performance of all algorithms does, in fact, decrease as weight variations increase, suggesting that as face recognition methods move towards unconstrained settings weight variation needs to be accounted for.

Finally, our experiments also show that methods based on correspondences perform better as the weight variation increases. While not developed specifically for face recognition with weight variation the correspondence-based algorithms perform solidly and are quite robust to this variation.

Chapter 5

Face Recognition with Large Pose

Variation

In this chapter we present a dynamic programming-based stereo method that accounts for slant and that turns out to be excellent at handling large pose variation.

5.1 Introduction

Our work presented in Chapter 3 has shown that stereo matching algorithms can be used to perform 2D face recognition in the presence of pose variation. In this approach, stereo is not used for reconstruction. Instead, two images are compared by matching them with a stereo algorithm and using the cost of this matching as a measure of similarity. This approach has produced the best current results on the pose variations found in the CMU PIE dataset.

However, this approach to face recognition stresses stereo matching algorithms significantly. When comparing faces taken from very different viewpoints, one essentially must perform stereo matching with a very wide baseline. While a great deal of progress has been made in wide baseline stereo [49], these approaches generally

do not produce a cost based on dense correspondences that is appropriate for image comparison and face recognition.

In this chapter we propose a new algorithm for wide baseline, dense stereo matching that capitalizes on two characteristics of the problem that arise in the context of face recognition. First, although large changes in pose do create significant occlusions in a face, they generally do not affect the monotonicity of correct matches. Even when matching a frontal view of someone to her profile, we can establish a continuous matching over one half of the face. This allows us to apply dynamic programming-based stereo algorithms that might be unsuitable for wide-baseline matching of more general scenes. Second, in wide-baseline stereo slant and tilt affect the appearance of an object. This creates a chicken-and-egg problem in which it is difficult to find the right match for image points without knowing the slant and tilt, but one needs correspondences to determine the slant and tilt. However, pose variation in faces tends to produce foreshortening primarily in the direction of the epipolar lines. We show that this allows us to use dynamic programming to solve for the main component of foreshortening at the same time that we find correspondences.

We have also included a curvature prior on our stereo matching algorithm. This seems to help in cases in which there is small variation in pose while accounting for slant seems to help in cases where the variation in pose is large.

We test the resulting stereo matching algorithm using the PIE dataset. We show that this method outperforms the approach presented in Chapter 3, as well as other previous approaches to face recognition with pose variation.

5.2 Stereo Matching with Slant

In Chapter 3 we have shown that stereo matching algorithms can be used to perform 2D face recognition in the presence of pose variation. In this approach, stereo is not used for reconstruction. Instead, two images are compared by matching them with a stereo algorithm and using the cost of this matching as a measure of similarity.

We hypothesize that for face recognition, slant alone has a very significant effect. This hypothesis is motivated by the observation that in face recognition, images are usually taken of upright people by upright cameras. Large variations in pose generally occur as the face turns from frontal towards profile. Therefore, epipolar lines relating two images tend to be approximately horizontal. At the same time, horizontal lines across a face tend to experience much greater depth variations than do vertical lines. Therefore, while the effects of tilt cannot be completely dismissed in face images, we expect that a stereo matching algorithm that accounts for slant alone can produce improvements in recognition performance when there are large pose variations. This is important because we will show that slant can be accounted for with a dynamic programming algorithm.

When two images are matched with a variation in lighting, it is important to somehow normalize the images to overcome the effects of local changes in intensity. In this work we focus on one of the most common approaches, in which we match small windows between images with intensities normalized to remove additive and multiplicative effects. This requires us to account for the effects of slant on window size, but other representations that we have examined seem to raise similar issues.

Next we examine the effect of slanted surfaces in window-based stereo matching [44]. We assume that the two images have been rectified so that the epipolar lines

are horizontal. Further, suppose that we use a window for matching that is an axial aligned rectangle in the left image. We consider which region in the right image will correspond to this rectangle.

First we note that each of the horizontal sides of the rectangle lie on a single epipolar line, and so they must lie on this same line in the right image. Next, we note that since the surface is slanted, $\frac{\partial d}{\partial v} \approx 0$. This means that the two left corners of the rectangle in the left image will have approximately the same disparity. The same will be true of the two right corners. This means that the region in the right image that corresponds to the rectangle in the left image will have two nearly vertical sides, and will also be approximately an axial aligned rectangle. The height of these two rectangles will be the same, since their top and bottom sides lie on the same two epipolar lines. However, the width of the two rectangles can differ significantly. This is because the slanted surface can cause different degrees of foreshortening in the two images. We illustrate this in Figure 6.2.

We can use a first order approximation to determine how this change in width depends on the change in disparity in the image. To do this, we need only consider one of the horizontal sides of the rectangle in the left image. Denote the upper left corner of this rectangle p_l , the upper right corner p_r , and a point halfway between the two as p_c . Denote the width of the rectangle $w = \|p_l - p_r\|$. Then, if we denote the disparity values at these three points as d_l, d_r, d_c , the width of the rectangle in the right image will be $w + d_r - d_l$. If we denote the change in disparity at p_c by d'_c then, to first order, we may say the rectangle in the right image will have a width of $w(1 + d'_c)$. In the next section we will use this as the basis of a dynamic programming matching algorithm.

Note that the expression above gives a negative width when $d' < -1$. This is correct, since in this case the order of d_l and d_r will be different in the two images. Such a situation violates the monotonicity constraint in matching.

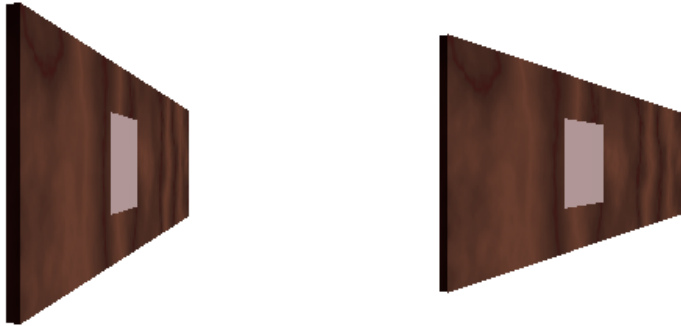


Figure 5.1: A wooden wall with a small patch marked seen from two viewpoints. This example illustrates the critical importance of handling slant correctly.

5.3 Dynamic Programming Algorithm

Dynamic programming (DP) approaches to stereo matching have been widely used [21, 22]. These are suitable for face recognition because they are fast and images of faces can be matched using a monotonicity constraint [5]. The chief disadvantage of using DP in stereo is inconsistency of matching across scan lines. While this produces artifacts in reconstruction, it does not significantly affect the matching cost, which is all that is used in face recognition.

DP matches one scan line at a time. We work in an $(x - d)$ space, in which x represents the location of a pixel along a scan-line in the left image, and d represents the disparity assigned to this pixel. This representation makes explicit the change in disparity as we move from one match to the next; the effects of slant on window size can be determined from this change in disparity. It also allows us to

use fractional values of d , which is important in calculating accurate windows. Even though our representation is asymmetric between the images, the costs of matching and occlusion are treated symmetrically.

5.3.1 The Algorithm

The core of our algorithm is to associate a disparity to every pixel in one scan line using dynamic programming. We can think of DP as the process of filling up a table T of possible x (position) and d (disparity) values. $T(x, d)$ gives the cost of the cheapest set of matchings and occlusions that account for all pixels in the left image up to pixel x , and all pixels in the right image up to $x + d$. x ranges from 1 to N , d ranges between minimum and maximum disparity values, and takes on fractional values. This allows subpixel matching, which has an important effect on window size. We proceed recursively by determining the minimum cost sequence of matchings that would result in matching pixel x with disparity d for each (x, d) pair, assuming that we have already computed this for all pairs that have a smaller value of x , or an equal value of x and a smaller value of d .

Finally, we define a curvature prior. Incorrect correspondences tend to have high total curvature. We implement the curvature prior using multiple tables or planes. Each plane stores costs for correspondences ending at a given slant, α . Jumping between planes incurs a cost that is proportional to the change in slant represented by each of the two planes. This type of prior was proposed by Belhumeur in his classical work on binocular stereopsis [5].

There are three types of moves that can be made in filling in a new table entry: matching moves, left occluding moves and right occluding moves. The table C_m is a 3-dimensional array that for each position (α, x, d) has the best cost to account

for pixels up to position x on the left scan line and up to position $x + d$ on the right scan line and ending in a match with a slant of α . Similarly, there are two two-dimensional occlusion tables (C_{ol} and C_{or}), that for a position (x, d) store the cheapest cost to account for pixels up to x on the left scan line and $x + d$ on the right scan line and in which the last action is to occlude on the left/right.

5.3.2 Matching Moves

If we arrive at the correspondence implied by (x, d) through matching, this means that pixel x in the left image is matched to point $x + d$ in the right image. This must be based on a previous table entry that account up to $(x - 1, d_p)$. The cost of the best matching move is:

$$C_m(\alpha, x, d) = \min_{d_p \in (d-3, d+1)} c((x - 1, d_p), (x, d)) + \min \begin{cases} \min_{\beta} \{(\alpha - \beta)^2 + C_m(\beta, x - 1, d_p)\} \\ C_{or}(x - 1, d_p) + \gamma \\ C_{ol}(x - 1, d_p) + \gamma \end{cases} \quad (5.1)$$

where $\tan \alpha = d - d_p$. c indicates the cost of a move, which will match this one new pixel in the left image to a number of pixels in the right image that depends on the number of integers between $x - 1 + d_p$ and $x + d$. For pixels that are matched in the right image, we can interpolate to find the non-integer location in the left image that they match.

The value of $c((x - 1, d_p), (x, d))$, then, is the sum of a matching cost that is computed for each pixel that is matched. Observe that in $c((x - 1, d_p), (x, d))$ the value d_p depends directly on the value of α . Another way of writing it would

be $c((x - 1, d_p(\alpha)), (x, d))$, but we don't do so to simplify notation. This cost is determined by the approximation to normalized SSD used by Criminisi et al.[22].

The formula of $\text{NSSD}(l, r)$ is:

$$\frac{1}{2} \left[\frac{\sum_{\delta \in \Omega} \left((I_{p_l+\delta}^l - \bar{I}_{p_l}^l) - (I_{p_r+\delta}^r - \bar{I}_{p_r}^r) \right)^2}{\sum_{\delta \in \Omega} (I_{p_l+\delta}^l - \bar{I}_{p_l}^l)^2 + \sum_{\delta \in \Omega} (I_{p_r+\delta}^r - \bar{I}_{p_r}^r)^2} \right] \quad (5.2)$$

where I^l is the left image and I^r is the right image and \bar{I} denotes the global mean of the image. In this method the ‘‘image’’ refers to 3×7 overlapping windows (or patches)¹.

The curvature prior is implemented as a penalty for changing slant planes. This can be observed from the $(\alpha - \beta)^2$ in Equation 5.3. Additionally, γ is a penalty for entering or leaving an occluded state.

When matching a pixel in the left image, we use $d' = d - d_p$ to determine the window size in the right image. We then use interpolation to create a matching window in the right image and resize it to be the same size as the window in the left image. The size of the window in the left image is fixed at 3×7 . The size of the window in the right image is therefore $3(1 + \tan \alpha) \times 7$

When matching a pixel in the right image, we interpolate the disparity for that match appropriately, so we can determine a point in the left image that matches it. We then similarly use interpolation to create an appropriate matching window in the left image. As discussed below, we only consider values for d_p for which $-1 < d' \leq 3$ since other values signal an occlusion.

¹We will abuse notation and define $\text{NSSD}(l, r)$ as NSSD as defined before in a 3×7 window around the points (l, s) and (r, s) of the images, where s is the current scan line.

5.3.3 Right Occluding Moves

In addition to matches, we allow for occlusions in either the left or right image. When there is an occlusion in the right image the disparity increases. In this case, the x value that indicates the position of the last pixel in the left image that has been accounted for does not change. The occlusion cost is based on the number of occluded pixels. That is:

$$C_{or}(x, d) = \min \begin{cases} \min_{\alpha} C_m(\alpha, x, d) + \gamma \\ \min_{d_p < d} (\lfloor d \rfloor - \lfloor d_p \rfloor)M + C_{or}(x, d_p) \end{cases} \quad (5.3)$$

where M is the cost of a single occlusion. The top part of the equation defines a cost to enter the occluded state from a matching state. The bottom part defines a cost to move along the occluded state.

5.3.4 Left Occluding Moves

If k pixels in the left image are occluded to reach (x, d) , this implies that previous to the occlusion we had accounted for $x - k$ pixels in the left image, with a disparity of $d + k$. Therefore, we have:

$$C_{ol}(x, d) = \min \begin{cases} \min_{\alpha} C_m(\alpha, x, d) + \gamma \\ \min_{k < x} Mk + C_{ol}(x - k, d + k) \end{cases} \quad (5.4)$$

Similar to right occluding moves, the top part of the equation defines a cost to enter the occluded state from a matching state. The bottom part defines a cost to move

along the occluded state. It is not possible to jump from occluding on the left to occluding on the right and vice-versa.

5.3.5 Total Cost for Recognition

Finally, we compute $T(x, d)$ as the cheapest of these possible moves, that is:

$$T(x, d) = \min_{\beta} C_m(\beta, x, d) \quad (5.5)$$

The cost of matching between two stereo pairs is therefore $\min_d T(N, d)$. Following [14] we note that in recognition, one does not know which image should be treated as left, and which should be right. Therefore, we try both possibilities, taking the one that produces a minimum cost. Furthermore, we can try flipping one of the images. This allows us to effectively match a right profile image to a left profile image, even though technically there may be no corresponding points visible in both images. Again, we use the flipped image only when this results in a lower cost matching. Similarly, we refer to flipped pose pairs as the cases where the azimuthal angles of the poses being compared have different signs, and unflipped is when the azimuthal angles have the same sign.

5.4 Experimental Evaluation

We have tested our algorithm using the CMU PIE dataset [70]. This dataset consists of 13 poses of which 9 have approximately the same camera altitude (poses: c34, c14, c11, c29, c27, c05, c37, c25 and c22). Three other poses have a significantly higher camera altitude (poses: c31, c09 and c02) and one last pose has a significantly

lower camera altitude (pose c07). Additionally, we consider 22 lighting conditions with lights on (called the *lights* track).

Thumbnails were generated using four hand-clicked points per face. This is enough to estimate the epipolar geometry under a scaled orthographic projection assumption. The height of the thumbnails is 72 pixels; the width is pose dependent. In our setup the images being matched have been rectified so that the epipolar lines are horizontal.

All results presented here are under gallery-probe experiments using the 68 individuals in the CMU PIE dataset. In this type of experiment a gallery is built using images with one pose and is queried with images in another pose. We will call a variation of more than 45° a large pose variation, and a variation of 45° or less a small pose variation.

A number of prior experiments have been done with pose variation using the CMU PIE database, but somewhat different experimental conditions. We will compare our results with our previous results using SMD (Stereo Matching Distance) [14]. That method produces the best published results across the 13 pose conditions in the CMU PIE dataset. Also, since our algorithm is similar to [14] except for our method of compensating for slant, this provides a direct evaluation of this innovation. The results for this comparison are presented in Table 7.2. We also compare with the method of Romdhani et al. [63] which is based on 3-D morphable models, a method that historically has had excellent performance in this type of task. There are several other works that focus on pose and illumination variation and evaluate on the CMU PIE dataset (see [85, 33, 17]). Most of them don't evaluate using large variation in pose [85, 17] and for the ones that do [33], the method of Castillo and Jacobs [14] has already been shown to produce significantly better performance.

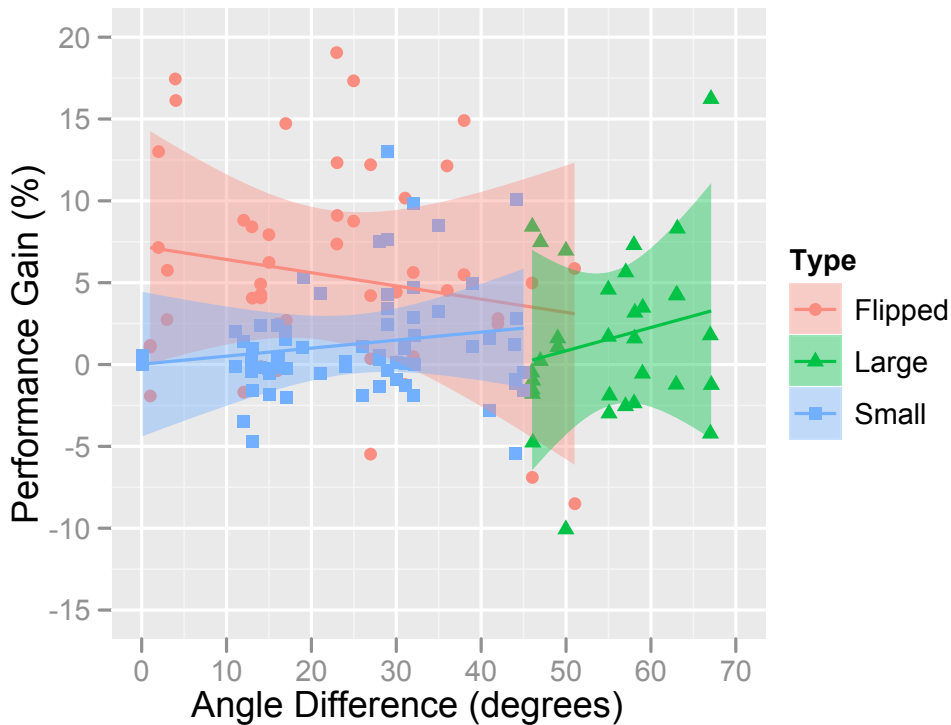


Figure 5.2: Gain in performance of Slant SMD compared to SMD (in the 68 face test case), as the angle difference changes. Flipped refers to cases where the azimuthal angles of the poses being compared have different signs, unflipped is when the azimuthal angles have the same sign. All bands show a 90% confidence interval.

Table 5.1: A comparison of recognition accuracy averaged across pose of our slant-compensated stereo matching distance with other methods.

34 Faces

Method	Accuracy
Eigenfaces [33]	16.6%
FaceIt [33]	24.3%
Eigen light-fields (Multi-point norm.) [33]	66.3%
SMD (Castillo and Jacobs [14])	86.8%
PLS (Sharma and Jacobs) [69]	90.1%
Slant SMD	90.1%

68 Faces

Method	Accuracy
LiST (Romdhani et al. [63])	74.3%
SMD (Castillo and Jacobs [14])	82.4%
Slant SMD	85.3%

Our experiments make the following points:

- Our method eliminates 16% of the errors made by state-of-the-art methods. Additionally, we show that this difference in performance is statistically significant.
- Our method is robust to simultaneous large variation of pose and illumination.

In the next two sections we will describe our experiments and our results.

5.4.1 Pose Variation Experiments

A summary of our pose variation experiments is presented in Table 7.2. These results show that overall our Slant SMD is better than SMD and that this increase in performance comes from being better at cases in which there is large pose variation.

The general behavior of the pose pairs can be analyzed in two cases: for flipped pose pairs the new slant-based method works significantly better than SMD at small pose variation and the relative performance gain decreases as the pose change increases. For unflipped pose pairs the slant-compensated method does not work better than SMD at low pose variation, but it becomes more useful as the pose variation increases. Details of this behavior can be seen in Figure 5.2 along with confidence intervals on the prediction.

Table 5.2 shows the details of both stereo methods across all pose variation cases studied. In this table, the pose pairs where there is large variation in pose are marked for comparison purposes.

Statistical Significance

To determine the significance of these results we used McNemar’s test [50]. We tabulated the two methods we wanted to compare (SMD and Slant SMD) with the dichotomous trait: correct/incorrect.

We are, therefore, performing a hypothesis test where the null hypothesis is that the probability that a face is classified correctly by SMD and incorrectly by Slant SMD is equal to the probability of a face being classified correctly by Slant SMD and incorrectly by SMD. The alternative hypothesis is that the probability that a face is classified correctly by SMD and incorrectly by Slant SMD is different from the probability of a face being classified correctly by Slant SMD and incorrectly by SMD.

We perform the test at individual cells (a given gallery and a given probe), over all galleries (all galleries for a fixed probe), over all probes (all probes for a fixed gallery), or over the entire table.

Globally, using McNemar’s test, we can establish that Slant SMD is significantly better than SMD ($p < 10^{-8}$, OR = 2.3). The details for individual cells can be seen in Table 5.2.

Table 5.2: Pose variation table for 68 faces comparing the use of the stereo matching method of [14] with our slant-compensated method. Cell format: $\langle \text{accuracy Slant SMD} \rangle / \langle \text{accuracy for SMD [14]} \rangle$. Pose pairs are labeled as follows: *: unflipped and pose variation less than 45° , †: unflipped and pose variation greater than 45° and ‡: flipped pairs. The diagonals are not included in any average. In cells with gray background, the performance gain is significant at 95% (McNemar’s test). The table layout is the same as [63] and [36].

azimuth altitude prb. pose gall. pose	-66 3 c34	-47 13 c31	-46 2 c14	-32 2 c11	-17 2 c29	0 15 c09	0 2 c27	0 1.9 c07	16 2 c05	31 2 c37	44 2 c25	44 13 c02	62 3 c22	avg
c34	-*/-*	93*/79*	96*/91*	87*/78*	72‡/65‡	54‡/38‡	46‡/44‡	43‡/26‡	56‡/50‡	62‡/50‡	72‡/60‡	76‡/71‡	72‡/56‡	68/59
c31	94*/91*	-*/-*	100*/99*	94*/96*	91*/94*	79‡/78‡	72‡/65‡	53‡/50‡	65‡/62‡	76‡/65‡	88‡/84‡	75‡/72‡	78‡/60‡	80/76
c14	97*/97*	97*/100*	-*/-*	100*/97*	99*/91*	90‡/87‡	88‡/79‡	69‡/71‡	79‡/79‡	82‡/76‡	66‡/59‡	81‡/76‡	81‡/78‡	85/82
c11	97*/94*	99*/97*	99*/99*	-*/-*	100*/100*	96*/97*	99*/94*	94*/94*	91‡/88‡	96‡/94‡	78‡/79‡	94‡/87‡	63‡/65‡	92/90
c29	76‡/87‡	99*/97*	100*/96*	100*/100*	-*/-*	100*/100*	100*/99*	99*/100*	97‡/96‡	99‡/94‡	76‡/82‡	87‡/81‡	46‡/53‡	89/90
c09	57‡/54‡	93‡/91‡	84‡/84‡	99*/99*	99*/100*	-*/-*	100*/100*	93*/97*	97*/94*	96*/94*	91‡/85‡	90‡/90‡	69‡/65‡	88/87
c27	56‡/60‡	90‡/93‡	90‡/91‡	100*/97*	100*/99*	99*/99*	-*/-*	100*/100*	100*/97*	100*/99*	96*/97*	99‡/97‡	60‡/62‡	90/90
c07	38‡/40‡	63‡/62‡	75‡/79‡	99*/97*	100*/100*	97*/96*	100*/100*	-*/-*	100*/100*	99*/99*	91*/88*	94‡/97‡	41‡/32‡	83/82
c05	62‡/71‡	82‡/79‡	94‡/90‡	93‡/93‡	99‡/97‡	97*/97*	99*/99*	100*/100*	-*/-*	100*/100*	99*/100*	100*/99*	78‡/78‡	91/91
c37	71‡/66‡	78‡/74‡	93‡/85‡	93‡/94‡	94‡/90‡	90*/91*	99*/97*	97*/99*	99*/100*	-*/-*	99*/100*	100*/100*	90*/91*	91/90
c25	84‡/65‡	79‡/79‡	69‡/56‡	75‡/66‡	71‡/71‡	82‡/85‡	85*/91*	90*/79*	97*/97*	100*/100*	-*/-*	99*/99*	96*/94*	85/81
c02	81‡/81‡	76‡/71‡	82‡/74‡	90‡/81‡	84‡/69‡	91‡/93‡	94‡/90‡	84‡/85‡	97*/93*	100*/100*	100*/99*	-*/-*	99*/99*	89/86
c22	75‡/57‡	71‡/62‡	81‡/66‡	66‡/56‡	49‡/44‡	56‡/49‡	51‡/47‡	40‡/35‡	74‡/66‡	87*/76*	94*/88*	99*/91*	-*/-*	70/61

5.4.2 Pose+Illumination Variation Experiments

Table 5.3 shows the results of the pose+illumination experiments performed. These experiments show the robustness of our method under front-to-profile comparison when there is also variation in illumination.

In this experiment images in two poses are compared and one of them (the gallery) is under lighting condition 12, the query is always in profile and illuminated in the lighting condition indicated in the table.

Our experiments show that our slant compensated method works considerably better than SMD under these conditions. The approach of [63] does still outperform both stereo-based methods. This may be because of the use of a 3-D morphable model and more sophisticated representations of the effects of lighting (at the same time, one should note that the decision to use lighting condition 12 for the gallery was made originally by [63] as one that is favorable for their method; [14] and we use the same gallery to allow comparisons). These results also suggest there is a lot of room for improvement for face recognition in this challenging setup.

Table 5.3: Pose + illumination variation for frontal, side and profile probe table using the stereo matching method of Castillo and Jacobs [14], the method of Romdhani et al. [63] and our slant compensated method. Gray background indicates significant difference between Slant SMD and SMD, the highlighted method is significantly better. F: front, S: side, P: profile.

G-P	Method	lighting condition																						avg
		2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22		
F-S	Slant SMD	80	88	95	94	98	98	100	100	94	100	100	100	100	97	95	92	95	95	100	100	100	96	
	SMD	92	95	94	98	100	98	100	100	98	100	100	100	100	98	97	95	98	100	100	100	100	98	
	List	60	78	83	91	89	92	94	97	89	97	98	97	98	97	94	89	85	86	97	98	97	91	
F-P	Slant SMD	36	35	42	47	57	50	55	55	42	60	64	52	58	48	39	41	41	47	60	60	52	50	
	SMD	32	35	36	35	41	36	44	44	29	45	52	45	47	45	39	35	36	38	44	50	50	41	
	List	22	28	45	65	65	65	48	57	58	72	78	77	83	80	71	75	58	54	72	58	78	60	
S-F	Slant SMD	88	86	89	94	98	92	100	100	92	100	100	100	100	98	98	97	94	95	100	100	96	96	
	SMD	85	92	95	98	100	98	100	100	98	100	100	100	100	100	100	98	98	100	100	100	98	98	
	List	50	84	85	94	94	96	99	100	93	97	99	100	99	97	99	91	88	88	99	100	97	93	
S-P	Slant SMD	50	39	33	44	58	48	64	72	42	76	82	79	82	73	75	72	44	57	75	82	77	63	
	SMD	41	41	35	47	57	52	55	64	48	60	70	63	66	54	51	48	38	54	63	64	57	54	
	List	31	26	47	65	65	71	75	71	71	82	91	91	91	93	84	87	60	57	75	81	90	71	
P-F	Slant SMD	27	26	23	38	54	41	42	50	27	57	57	47	48	47	41	44	29	48	58	60	48	43	
	SMD	26	29	33	38	38	29	35	39	32	47	48	44	44	36	39	32	35	42	48	47	38	38	
	List	29	63	54	51	65	57	63	69	60	66	75	82	82	81	87	76	43	51	59	74	74	64	
P-S	Slant SMD	29	32	27	33	45	42	60	69	38	66	75	75	79	72	64	60	35	48	64	72	75	55	
	SMD	17	16	23	25	42	35	42	47	27	48	55	48	58	60	52	44	22	44	51	57	60	42	
	List	49	54	51	53	62	65	78	88	60	75	78	85	90	90	93	85	47	50	71	79	90	71	

5.4.3 Discussion

Our experiments show that our method outperforms existing methods for large pose variation. There is a small fall-off compared to our previous method [14] when the poses are very similar (small variation in pose). The method also works well in small pose variation cases.

5.5 Conclusions

Dense, wide-baseline stereo matching is a very challenging problem. However, when we are using stereo matching for face recognition, our problem is somewhat simplified. Faces, even seen from quite different viewpoints, can be matched monotonically, making it practical to apply dynamic programming. Furthermore, we hypothesize that the effects of slant predominate over those of tilt, due to the shape and typical imaging conditions for faces. This allows us to develop a dynamic programming-based stereo matching algorithm that fully accounts for the effects of slant on window size. This leads to significant performance gains in face recognition in the presence of large pose variations.

Chapter 6

Towards Dense, Wide-baseline Stereo under Varying Illumination

In this chapter we present our MRF-based stereo matching formulation that fully accounts for slant and tilt using a deformed-window unary cost and a novel pairwise cost that measures curvature.

6.1 Introduction

Originally, stereo focused on the problem of matching images taken at the same instant using a narrow baseline. However, interest has grown in matching images taken from very different viewpoints at different times. This can enable reconstruction of scenes using images taken from mobile platforms (e.g., Google street maps), visualization using internet images (e.g., phototourism [71]) or even face recognition (e.g., [15]). Matching such images presents new challenges because they can have much wider baselines and significant lighting variations.

These two challenges are intertwined. A great many approaches have been suggested for image matching with lighting variation; these often include the use of

image gradient directions, the use of windows or region-based descriptions, which allow for normalization. However, foreshortening from changing viewing direction affects both the direction of image gradients and the shape and size of corresponding windows. Wide baselines can cause these effects to be quite large.

To address this problem we propose a stereo matching approach in which pixels are labeled according to their disparity and relative slant and tilt. This allows us to compare pixels in different images using windows that are rectified to allow for changes in window shape due to viewpoint change. A key contribution is the proposal of a new pairwise cost function that measures the consistency between neighboring labels. This pairwise cost is a metric, allowing us to use Graphcuts to optimize the resulting cost.

Representing disparity, slant and tilt leads to a potentially huge label set. However, each label represents a planar surface; consequently a relatively modest number of labels are needed to accurately approximate any given scene. We exploit this using an algorithm in which we incrementally add labels as needed, so that our Graphcuts problem remains manageable.

Our cost can be adapted to any stereo matching method that uses windows or regions when comparing pixels. We experiment using a very simple but popular approach of comparing windows by the sum of square differences between their normalized intensities. Our experiments focus on showing that stereo matching using slant and tilt can provide a substantial improvement over matching that uses only fixed sized windows, or over pixel based matching. We do this using our own new dataset of outdoor images taken with wide baselines and lighting variation, using face images with varying pose and lighting, from the CMU PIE [70] dataset

and also using the wide baseline images from the DAISY [76] dataset. We also show experiments on the standard Middlebury data set [67].

6.2 Previous Work

We focus on a Markov Random Field (MRF) formulation of stereo. Many recent papers have proposed, effective optimization algorithms for use in stereo matching using MRFs (eg., belief propagation [72] and QPBO-I [64]). When pairwise costs between pixels obey a regularity, or metric constraint, Graphcuts-based methods [12] have proven extremely effective and efficient.

Our work builds on work on image matching with lighting variation, both for stereo and object recognition. Inspired especially by [47], many recent approaches have converged on representations that use histograms of image gradient directions. Recently, a related representation has been shown to be extremely effective in wide baseline stereo matching [76]. We use simple sum-of-squared distances (SSD) between two image regions after normalizing them. This has been used in stereo, for example, by [22].

The effect of slant and tilt on window-based matching was discussed in [23, 44]. When the baseline is not wide, the goal of these approaches is to compensate for small changes in foreshortening, (eg., [56]) or to use the subtle effects of foreshortening to perform matching with subpixel accuracy. [45] propose a belief propagation-based framework for stereo matching in the presence of slanted and curved surfaces. Also, [15] proposes a dynamic programming method that accounts for slant (but not tilt) in wide baseline matching of faces. There is also work that uses a second

order prior for MRFs for stereo [83] to remove the fronto-parallel bias of existing stereo methods.

We represent the relative slant and tilt of a surface at each pixel, in terms of the horizontal and vertical changes in disparity. Previous approaches, [10] for example, labels patches of images with planar surfaces or b-splines in the scene.

6.3 Stereo Matching Cost

In this section we introduce our stereo matching cost. First, we define a grid-shaped graph, with one labeled node for each pixel in the first image, and edges between horizontal and vertical neighbors. Next we describe the unary costs for these labels, based on normalized SSD with deformed windows. Finally, we define a pairwise cost for the labels of two adjacent pixels. This cost essentially measures the amount of curvature implied by this pair of labels. We formulate a binary cost that is a metric, allowing us to optimize the complete stereo matching cost using Graphcuts.

The cost to minimize is therefore:

$$E(f) = \sum_{p \in S} U(p, f_p) + \lambda \sum_{\{p,q\} \in N} d_{p,q}(f_p, f_q) + \gamma \sum_{l \in \mathbb{L}} \delta_l(f) \quad (6.1)$$

where S is the set of sites, p and q are sites, N is the set of neighboring sites and f is a labeling, in which f_r is the label assigned to site $r \in S$ under the labeling f . U is the unary cost function and d is the binary or pairwise cost function. We also define $\delta_l(f)$ as:

$$\delta_l(f) = \begin{cases} 1 & \exists p : f_p = l \\ 0 & \text{otherwise} \end{cases} \quad (6.2)$$

This assigns a fixed cost to every distinct label used. This discourages the use of excess labels, which, as we will see, increase the computational cost of using Graphcuts.

6.3.1 Labels

Disparity determines the location of matching pixels, while slant and tilt determine the shape of windows. Therefore we label each pixel with a triple of parameters, (s, t, c) , that describe a plane in *disparity space*. Calling disparity w , this plane is given by $w = sx + ty + c$. For example, if we label a pixel at location (x_0, y_0) with (s, t, c) then the disparity at this pixel is given by $sx_0 + ty_0 + c$. In this representation, s encodes what we call the slant of the surface, given by $\frac{\partial w}{\partial x}$, while t represents the tilt, $\frac{\partial w}{\partial y}$. Slant and tilt are therefore defined in a disparity space, rather than the geometric space of the scene. This has the advantage that we can use this approach for matching when the epipolar geometry of an image pair is known, but the magnitude of the baseline is still unknown. We adopt this approach rather than explicitly using disparity as a label because we wish two neighboring pixels to be coplanar if and only if they have the same label.

6.3.2 Unary Cost

We now examine the effect of slanted and tilted surfaces in window-based stereo matching [56, 44]. We assume that the two images have been rectified so

that the epipolar lines are horizontal. Also, in the left image all matching windows will be axis-aligned rectangles (extension to other shapes is straightforward). To compute the unary cost we determine the shapes of matching windows in the right image.

To analyze the deformation of the window, consider an $l_x \times l_y$ rectangle with corners: (x_l, y_t) , (x_r, y_t) , (x_l, y_b) , (x_r, y_b) as shown in Figure 6.1. These points on the left image will match the following points on the right image: $(x_l + w_{tl}, y_t)$, $(x_r + w_{tr}, y_t)$, $(x_l + w_{bl}, y_b)$ and $(x_r + w_{br}, y_b)$. Suppose also that the central point of this rectangular window, c , has disparity w_c , and the disparity horizontally is changing at a rate s and vertically is changing at a rate t . We can therefore establish that up to a first order approximation: $w_{tl} = w_c - \frac{1}{2}sl_x - \frac{1}{2}tl_y$, $w_{tr} = w_c + \frac{1}{2}sl_x - \frac{1}{2}tl_y$, $w_{bl} = w_c - \frac{1}{2}sl_x + \frac{1}{2}tl_y$, and finally $w_{br} = w_c + \frac{1}{2}sl_x + \frac{1}{2}tl_y$. The slant therefore affects the width of the window and the tilt affects the angle of the axis-aligned quadrilateral. We illustrate this in Figure 6.2.

In our experiments we use normalized SSD (NSSD). A rectangular window is deformed to account for slant and tilt in the right image. Most image descriptors

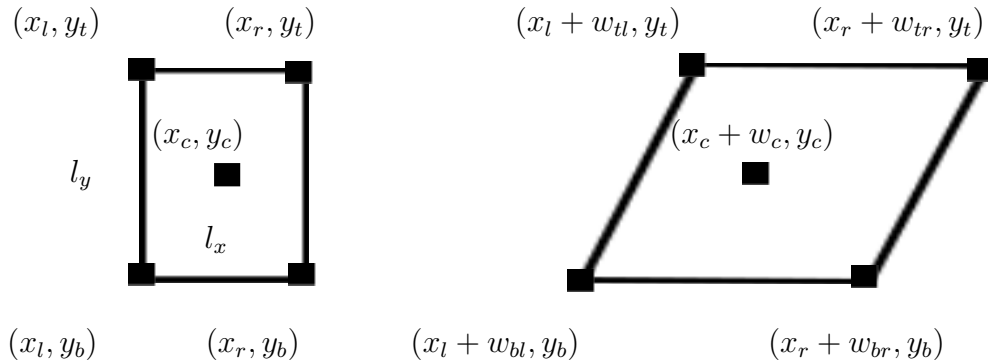


Figure 6.1: Deformation of the matching window under slant and tilt. See text for the values of w_{tl} , w_{tr} , w_{bl} , w_{br}

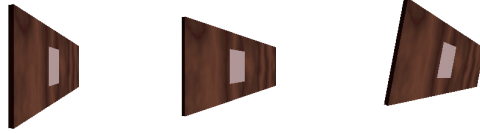


Figure 6.2: A wooden wall with a small marked patch as seen from three distinct viewpoints. This example illustrates the critical importance of handling slant and tilt correctly.

such as: normalized correlation, SIFT [47], GLOH [52], and DAISY [76] are window-based, and can be used with our method.

6.3.3 Pairwise Cost

We now describe a pairwise cost that can be applied to the labels of two neighboring nodes in the MRF, corresponding to neighboring pixels in the image. Each label implicitly specifies a point and normal direction in the x - y - w space. We can think of this as specifying a point on a surface, but this is a surface in an artificial disparity space rather than in the 3D scene. Intuitively, we imagine interpolating between two points on the surface associated with neighboring pixels. Our cost is the amount of curvature in this interpolated surface. Similar costs that seek to minimize the curvature in a reconstructed surface have long been used in stereo (eg., [32, 5]). However, these approaches have not formulated a cost based on curvature that is suitable for use with Graphcuts.

$d_{p,q}(f_p, f_q)$ is the cost for vertices p, q where f_p and f_q are labels for nodes p and q . This cost is defined only if p and q are neighbors. Note that the cost varies from one neighborhood to another. This is important, because the same labels can give rise to quite different geometric relationships at different nodes.

A node's label describes the plane $w = sx + ty + c$, from which we can determine the disparity at that node, which depends on the x - y coordinates of the node. s and t give the slant and tilt explicitly. Slant and tilt give us the surface normal, which is a unit vector in the direction $(s, t, 1)$, and so depends only on the label. We may refer to the normal vectors associated with labels f_p and f_q as n_p and n_q .

We will describe our metric using two distances that form components of the metric. First, we define $d_{\perp}(f_p, f_q) = \arccos(n_p \cdot n_q)$: $d_{\perp}(f_p, f_q)$ is a lower bound on the curvature along any path connecting any two points with these labels. However, it is not a good cost function, because this bound can be loose. For example, suppose $n_p = n_q = (0, 0, 1)$, but the disparity at p is 0 and at q is 20. $d_{\perp}(f_p, f_q) = 0$, but the two points can only be connected by paths with a great deal of curvature.

We therefore distinguish between two pairwise situations. If the node-label pairs (p, f_p) and (q, f_q) imply surface points in disparity space that can be connected with a surface containing no inflexion points, we call this a *reliable pair*. Otherwise, we call (p, f_p) and (q, f_q) an *inflexion pair*. For a reliable pair, $d_{\perp}(f_p, f_q)$ tells us the amount of curvature of an interpolated surface between the two points. Figure 6.3 shows inflexion and reliable pairs.

To define this distinction more precisely, suppose neighboring nodes p and q have labels f_p and f_q . Let p have image coordinates (x_1, y_1) and disparity w_1 , and let q similarly correspond to the point (x_2, y_2, w_2) in disparity space. Let v denote the vector $(x_2, y_2, w_2) - (x_1, y_1, w_1)$. Then if $v \cdot n_p \leq 0$ and $-v \cdot n_q \leq 0$ we say that the two points are convexly related. When $v \cdot n_p \geq 0$ and $-v \cdot n_q \geq 0$ we say the points are concavely related. The points form a reliable pair when they are either convexly or concavely related. Otherwise, they form an inflexion pair.

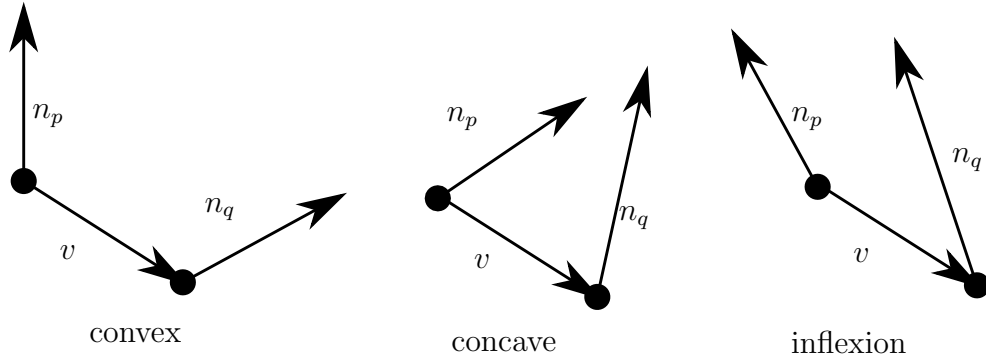


Figure 6.3: A simple example illustrating the inflexion and relatable relation. Relatable pairs are either convex or concave.

It is straightforward to show that d_{\angle} is a metric, and our primary goal is to adopt a distance that has this value for pairs of labels with a relatable relationship. However, we must also extend this distance to inflexion pairs in a way that obeys symmetry and the triangle inequality. We will define a distance for inflexion pairs that consists of the distance between a sequence of relatable pairs that connects them.

$$d_{I,p,q}(f_p, f_q) = \min_{c_1, \dots, c_n} \left(d_{\angle}(f_p, c_1) + \left(\sum_{i=1}^{n-1} d_{\angle}(c_i, c_{i+1}) \right) + d_{\angle}(c_n, f_q) \right) \quad (6.3)$$

where (c_1, \dots, c_n) is a sequence of labels of any length, provided that for all labels, (p, c_i) and (q, c_{i+1}) have a relatable relationship, and that (p, f_p) and (q, c_1) and (p, c_n) and (q, f_q) also have relatable relationships. $d_{I,p,q}$ captures the amount of curvature required to connect two points through a series of intermediate points (see Figure 6.5). Note that with a discrete set of labels we can compute $d_{I,p,q}$ using a shortest path algorithm. Note that $d_{I,p,q}$ is not symmetric, as illustrated in Figure

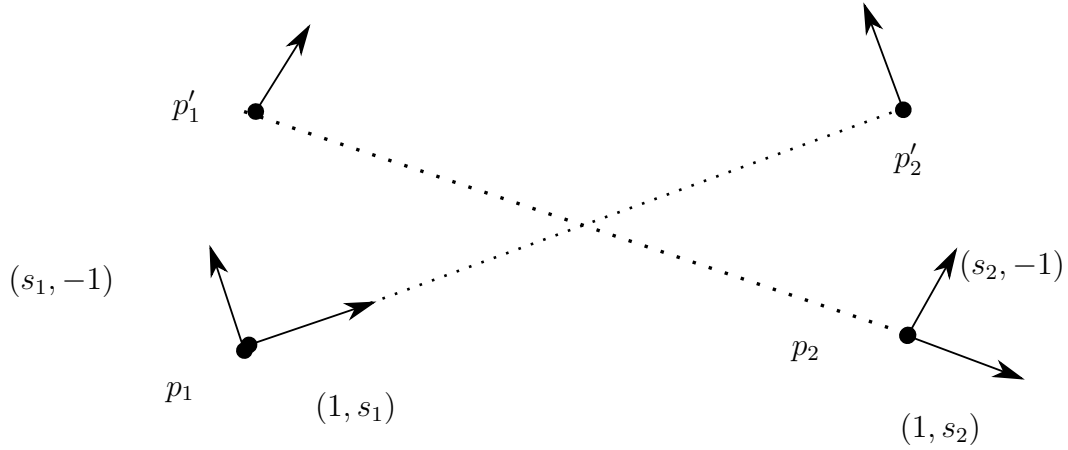


Figure 6.4: An example showing the symmetry of d_{\angle} .

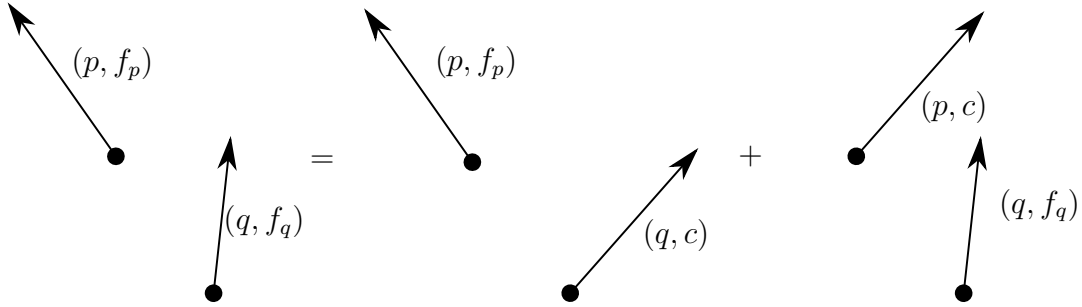


Figure 6.5: $d_{p,q}(f_p, f_q)$ might be computed as $d_{\angle}(f_p, c) + d_{angle}(c, f_q)$ for an appropriate c .

6.6. We therefore symmetrize our distance by defining:

$$d_{p,q}(f_p, f_q) = \begin{cases} d_{\angle}(f_p, f_q) & \text{for a relatable pair} \\ \min(d_{I,p,q}(f_p, f_q), d_{I,p,q}(f_q, f_p)) & \text{for an inflexion pair} \end{cases} \quad (6.4)$$

We note that our distance imposes symmetry on inflexion pairs in a somewhat artificial way. However, correct disparity maps of scenes generally contain few inflexion

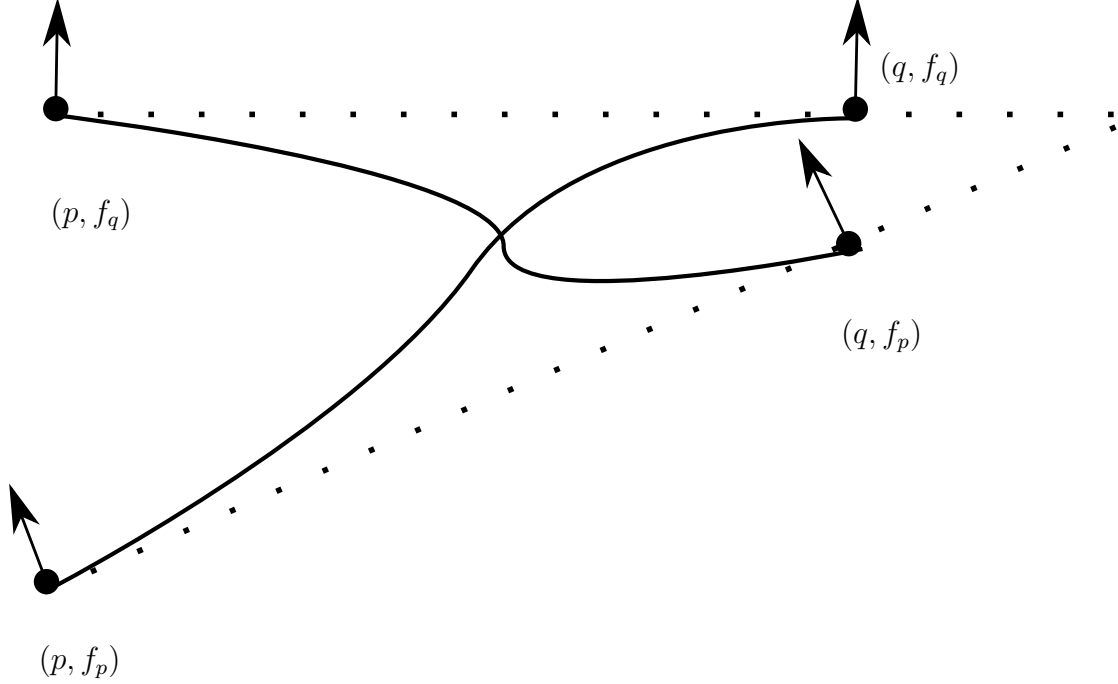


Figure 6.6: Exchanging labels does not produce a symmetric change in geometry, although in the example it does not greatly affect the curvature of an interpolating curve.

points, and our distance imposes a reasonable, high cost that effectively discourages spurious inflexions.

We now explain why this distance obeys the conditions of a metric. $d_{p,q}$ is a metric provided:

1. $d_{p,q}(f_p, f_q) = 0$ if and only if $f_p = f_q$.
2. $d_{p,q}(f_p, f_q) = d_{p,q}(f_q, f_p)$
3. $d_{p,q}(f_p, f_q) \leq d_{p,q}(f_p, c) + d_{p,q}(c, f_q)$ for any c .

Condition (1): First, if $f_p = f_q$ then the surface normals n_p and n_q are equal, and $d_{\perp}(f_p, f_q) = 0$. Moreover, the planes determined by these labels are identical, so that they create a relatable pair and $d_{p,q}(f_p, f_q) = d_{\perp}(f_p, f_q) = 0$. Next, we show that if $d_{p,q}(f_p, f_q) = 0$ then $f_p = f_q$. First, if $d_{p,q}(f_p, f_q) = 0$ then it must be the

case that $n_p = n_q$, and the planes in disparity space associated with these labels are parallel. If the planes are identical, then the labels are also identical. If the planes are parallel but not identical, then the labels form an inflexion pair. Note that in this case $d_{p,q}(f_p, f_q) = 0$ only if there exists a label c (or sequence of labels) such that $d_{\angle}(f_p, c) = 0$ and $d_{\angle}(c, f_q) = 0$, and for which these are relatable pairs. This can occur only if $c = f_p$ and $c = f_q$, so that $f_p = f_q$. The same reasoning holds for a sequence of intermediate labels.

This proof explains our decision to label each node with a plane in disparity space, rather than making disparity an explicit part of the label. In order to assign zero cost to planar regions, we must have labels that are identical if and only if the reconstructed scene points are coplanar.

In order to show that $d_{p,q}$ is symmetric, the main issue is to show that the inflexion relationship is symmetric. That is, if (p, f_p) and (q, f_q) have an inflexion (resp. relatable) relationship, then (p, f_q) and (q, f_p) also have a inflexion (resp. relatable) relationship. Given this, symmetry follows directly from the definitions of d_{\angle} and $d_{p,q}$.

To show that the inflexion relationship is symmetric, we first must consider how exchanging labels between two nodes affects the points associated with them in disparity space. First, we denote the plane and normal associated with f_p as P and n_p , and similarly define Q and n_q . As before, we denote the point in disparity space associated with (p, f_p) as $p_1 = (x_1, y_1, w_1)$, and similarly denote the point associated with (q, f_q) as $p_2 = (x_2, y_2, w_2)$. We also need to denote the points that occur if we exchange labels, so we call the point associated with (p, f_q) : $p'_1 = (x'_1, y'_1, w'_1)$, and similarly define $p'_2 = (x'_2, y'_2, w'_2)$.

Without loss of generality, we assume that p and q are horizontal neighbors, so that $y_1 = y_2$ and $x_2 = x_1 + 1$. Note that p'_1 (and p_2) lie in the $y = y_1$ plane, and also lie on the plane Q . These two planes intersect in a line whose slope in the x - w plane we denote by s_2 . p'_1 and p_2 are both on this line, and we have:

$$p'_1 = p_2 - (1, 0, s_2) = (x_1, y_1, w_2 - s_2) \quad (6.5)$$

Similarly,

$$p'_2 = (x_2, y_2, w_1 + s_1) = (x_1 + 1, y_1, w_1 + s_1) \quad (6.6)$$

We will now show that if (p, f_p) and (q, f_q) have a convex relationship, then (p, f_q) and (q, f_p) have a concave relationship. Denote the vector $v = p_2 - p_1 = (1, 0, w_2 - w_1)$ and $v' = p'_2 - p'_1 = (1, 0, w_1 + s_1 - w_2 + s_2)$. (p, f_p) and (q, f_q) have a convex relationship if and only if:

$$v \cdot n_p \leq 0 \quad \text{and} \quad -v \cdot n_q \leq 0 \quad (6.7)$$

We note that v lies in the $y = 0$ plane, so these equalities do not depend on the y component of n_p or n_q . Moreover, the projections of n_p and n_q into the $y = 0$ plane are parallel to $(s_1, 0, -1)$ and $(s_2, 0, -1)$ respectively. So (p, f_p) and (q, f_q) have a convex relationship if and only if:

$$\begin{cases} (1, 0, w_2 - w_1) \cdot (s_1, 0, -1) \leq 0 \\ -(1, 0, w_2 - w_1) \cdot (s_2, 0, -1) \leq 0 \end{cases} \quad (6.8)$$

Some algebraic manipulation then shows that this is equivalent to the conditions for (p, f_q) and (q, f_p) having a concave relationship, which are:

$$\begin{aligned} v' \cdot n_q \geq 0 &\equiv (1, 0, w_1 + s_1 - w_2 + s_2) \cdot (s_2, 0, -1) \geq 0 \\ -v' \cdot n_p \geq 0 &\equiv -(1, 0, w_1 + s_1 - w_2 + s_2) \cdot (s_1, 0, -1) \geq 0 \end{aligned} \quad (6.9)$$

Identical reasoning shows that exchanging the labels on concave points creates convex points. Therefore, if two points have an inflexion relationship, exchanging labels maintains that inflexion. Symmetry then follows directly from the definition of our distance.

Finally, we consider the triangle inequality. We need to show that $d_{p,q}(f_p, f_q) \leq d_{p,q}(f_p, c) + d_{p,q}(c, f_q)$ for any c . First, we note that $d_{\angle}(f_p, f_q)$ obeys the triangle inequality. This implies that $d_{p,q}(f_p, f_q) \geq d_{\angle}(f_p, f_q)$; if (p, f_p) and (q, f_q) have a relatable relationship, $d_{p,q}(f_p, f_q) = d_{\angle}(f_p, f_q)$; otherwise, $d_{p,q}(f_p, f_q)$ is defined through a sum of distances that all obey the triangle inequality. Next, suppose that (p, f_p) and (q, f_q) have a relatable relationship. From these two facts it follows that $d_{p,q}(f_p, f_q) \leq d_{p,q}(f_p, c) + d_{p,q}(c, f_q)$ for any c .

This leaves the case in which (p, f_p) and (q, f_q) have an inflexion relationship.

We can write

$$\begin{aligned} d_{p,q}(f_p, c) + d_{p,q}(c, f_q) &= d_{\angle}(f_p, a_1) + \cdots + d_{\angle}(a_m, c) + \\ &\quad d_{\angle}(c, b_1) + \cdots + d_{\angle}(b_n, f_q) \end{aligned} \quad (6.10)$$

where all pairs of labels on the right hand side for which we compute their d_{\angle} have a relatable relationship when applied to nodes p and q . Note that the sequence of

a_i or b_i might be empty. Clearly, then, from the definition of $d_{p,q}$ we have:

$$d_{p,q}(f_p, f_q) \leq d_{\angle}(f_p, a_1) + \dots + d_{\angle}(a_m, c) + d_{\angle}(c, b_1) + \dots + d_{\angle}(b_n, f_q) \quad (6.11)$$

since the sequence of intermediate labels $(a_1, \dots, a_m, c, b_1, \dots, b_n)$ is also a valid sequence of labels to use in computing $d_{p,q}(f_p, f_q)$.

In practice we can compute the cost directly instead of using a shortest path algorithm by creating in the inflexion cases a new label f_m that contains v and $v \times n$ where n is a vector that points in the direction of $n_p + n_q$. We then compute: $\min(d_{\angle}(f_p, f_m) + d_{\angle}(f_m, f_q), d_{\angle}(f_q, f_n) + d_{\angle}(f_n, f_p))$. We call this the analytic version of the cost. Considering that the analytic cost is both more accurate and faster to compute there is no reason to use the discrete (shortest paths) version of the cost.

6.3.4 Labels and GraphCuts

In implementing our method we face a trade-off; using a large number of labels allows us to represent slant and tilt accurately, but leads to a more expensive algorithm. However, coplanar surfaces in the scene should share the same label, suggesting that a final solution to the stereo problem should not require a large number of labels. We do not know a priori which labels will be needed, so we adopt an incremental solution in which we solve the stereo problem with a small number of labels and then incrementally add labels as needed.

We initialize by solving the stereo matching problem using only labels with zero slant and tilt. Then, at each iteration, we add labels that are near the labels used in the current solution. If a pixel has been labeled with one disparity, slant and

tilt, we add labels at that node that produce the same disparity along with values for the slant and tilt that differ from the current values by some delta. We begin with a delta of 30° , which we reduce by a factor of two at each iteration, allowing us to produce labels that are up to 45° from frontal. In order to discourage Graphcuts from finding solutions with unnecessary labels, we also include a penalty based on the number of different labels used.

6.3.5 Segmentation Cost

We also exploit a prior on the segmentation of the reference image [10].

$$s_p^{p,q}(f_p, f_q) = \begin{cases} v_h & \text{if same}(p, q) \text{ and } f_p \neq f_q \\ v_m & \text{if difference}(p, q) \text{ and } f_p = f_q \\ 1 & \text{otherwise} \end{cases} \quad (6.12)$$

where same and difference determine whether p and q fall on the same segment. We compute the segmentation of the reference image using EDISON [51]. We then multiply the segmentation cost by the pairwise cost. Observe that as the segmentation is constant, multiplying by the segmentation cost does not affect the metric property of the pairwise cost.

6.3.6 Discussion

To summarize, our distance is motivated by the desire to use slant, tilt and disparity in the unary cost in order to compensate for the effects of foreshortening on window size and shape. We also want a pairwise cost that discourages curvature. To impose zero cost on planar surfaces we must use labels that explicitly describe planes

that are tangent to the reconstructed surface, representing disparity implicitly. We then choose a cost that exactly measures curvature for labels that create surfaces that are locally convex or concave. For other labels, we use a cost that obeys the triangle inequality while also capturing the curvature of possible interpolating surfaces that contain inflexions. However, this curvature is inherently asymmetric when interpolation requires inflexion points, and so we artificially symmetrize the cost.

6.4 Experimental Evaluation

In this section we evaluate our method with real data from three sources:

1. The CMU-PIE [70] dataset, and our experiments with this dataset are referred to as PIE experiments. This dataset allows us to compare faces captured simultaneously under different illuminations.
2. POVRAY rendered stereo pairs with varying illumination. We call this series of experiments POVRAY [1]. Many papers evaluate on the Corridor POVRAY images; we extend the image set to contain varying illumination.
3. Our own outdoor images of a building which have a wide baseline and illumination variation. We call this series of experiments Outdoor. This dataset illustrates one of the main applications we envision for our method.
4. The dataset distributed by the authors of the DAISY descriptor [76]. We call this series of experiments DAISY. This dataset will allow us to compare our method with DAISY.
5. The Middlebury Stereo dataset.

In our experiments we evaluated four methods. We will briefly describe each:

- **GC+L2:** An L2 distance with direct pixel comparison, a Pott’s smoothness term, and a Graphcut based stereo matcher. We also tried the Birchfield and Tomasi [8] distance, however the difference was very small compared to L2 on these cases. We settled on L2 due to simplicity.
- **Second order prior (2op):** is the second order prior method of Woodford, et al. [83].
- **GC+NSSD:** NSSD data term on 3x7 windows, with a Pott’s smoothness term, and a Graphcut based stereo matcher. This is a natural baseline because it is identical to our method without slant, tilt and the pairwise cost.
- **Slant+Tilt:** our new method

In all output images red pixels are occluded. We now describe the results for each of the experiments.

6.4.1 PIE Experiments

We evaluated our method using the CMU PIE dataset. The images were rectified using hand-clicked points. Figure 7.3 shows one such example from the CMU PIE dataset. In this experiment we took pairs of images of the same individual in two different poses separated by 60 degrees (front to profile, PIE poses c29 to c27). For 8 front-profile pairs we generated hand-clicked ground truth disparity maps.

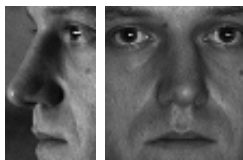


Figure 6.7: Two images from the CMU PIE dataset.

We then generated the disparity maps under the 22 illuminations in the PIE lights dataset. In a method robust to change in illumination the disparities should not change as the illumination varies. We have partitioned these illumination conditions into 5 groups. The first, G0, contains no lighting variation; we partitioned the remaining images into four groups from easiest (G1) to hardest (G4). Figure 6.8 shows the results of this experiment. First, we observe that of the three methods the best across all illumination conditions is Slant+Tilt. We note that even when there is no change in illumination the number of incorrect pixels is relatively large. This is due to the very large baseline. Also, in all cases, as expected, the methods degrade as the illumination conditions get more difficult.

6.4.2 POVRAY Experiments

We generated variations of the Corridor POVRAY pair under varying illumination using MRTStereo [1]. The MRTStereo tool generates stereo pairs and the

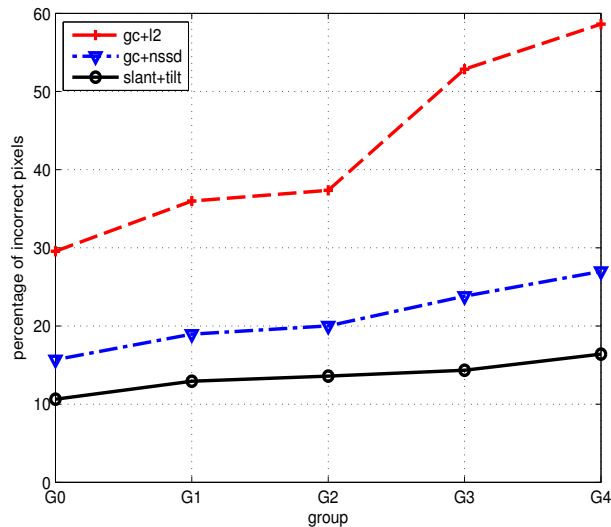


Figure 6.8: Results of the PIE experiment. Where the x axis is the group and the y axis is the average number of pixels per image labeled incorrectly according to the ground truth.

corresponding disparity maps. The same Corridor POV-Ray pair has been generated in Li and Zucker [45] and Woodford et al. [83].

The Corridor POV-Ray pair has 6 light sources. We have generated 4 test cases as follows: left image with 6 lights on vs right image with 6, 5, 4 and 3, lights on. Figure 6.9 shows a pair of images from the POV-Ray pair.

In this experiment we compare with the second order prior method of Woodford, et al. [83] and with GC+NSSD. The second order prior method (2op) while similar to ours in that it has no fronto-parallel bias, has no provision to handle illumination change and as expected does not perform well in such situations.

6.4.3 Outdoor Experiments

We evaluated our method on outdoor scenes of buildings. The images were rectified using hand-clicked points [28], results are presented in Figure 6.11. This experiment shows how our method robustly handles variation in illumination in a wide baseline setting, even under harsh illumination conditions.

Note that since the entire scene is visible from both viewpoints the correct correspondences have almost no occlusions. Also, the disparities should uniformly increase from left to right, and then decrease again.

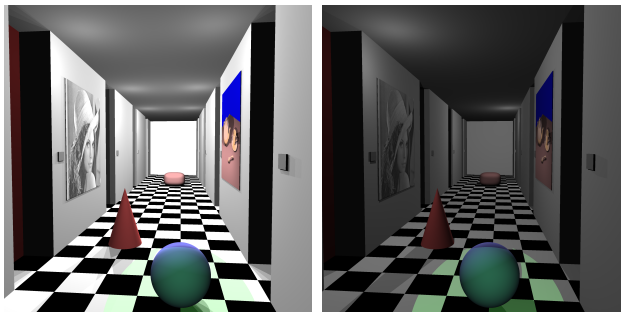
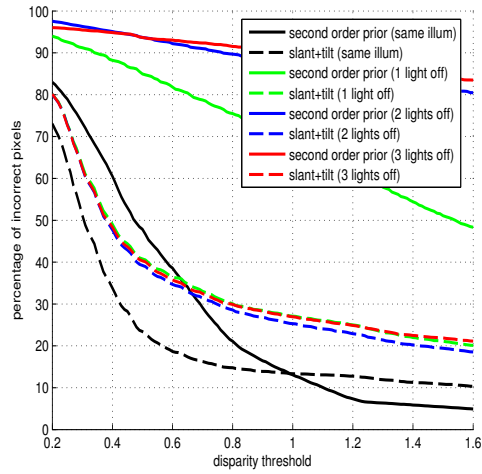
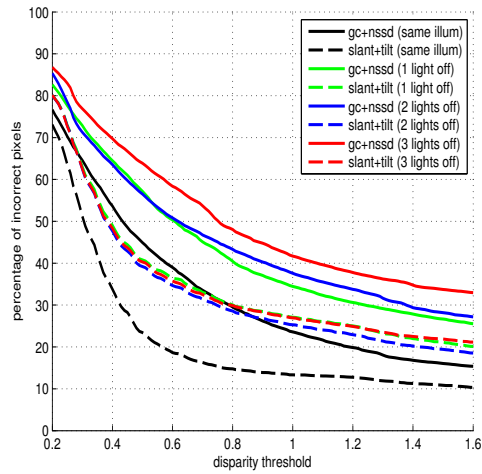


Figure 6.9: A left-right pair from our corridor image set across illumination.



(a)



(b)

Figure 6.10: Results of the POVRAY experiment. (a) comparison of our slant tilt method with the second order prior method of Woodford, et al. across illumination. (b) comparison of the slant+tilt method with gc+nssd.

These experiments suggest that wide baseline matching of outdoor scenes with varying illumination requires effectively handling slant and tilt. In this experiment our slant and tilt method clearly improves over gc+nssd.








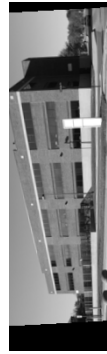








	Easy	Medium	Hard	Very hard
Right				
Left				
GC+NSSD				
Slant+Tilt				

Figure 6.11: Evaluation of our method under illumination variation. The first two rows show original images, the bottom three rows show disparity maps. Red pixels are occluded.

6.4.4 DAISY Experiments

We also evaluated our method using the dataset captured in the DAISY paper [76], which also addresses wide baseline matching. DAISY uses descriptors designed for wide baseline matching (unlike NSSD) and has particularly sophisticated handling of occlusion. We can see in Figure 6.12 that DAISY indeed handles occlusion well, although it is also the case that our new algorithm is more accurate in some portions of the image. We find these results to be very encouraging since our work addresses issues largely orthogonal to those discussed in [76].

6.4.5 Middlebury Stereo Experiments

We have evaluated using the Middlebury Stereo dataset. While our method is designed to work under wide-baseline setups with varying illumination, it does not fall apart in narrow-baseline and constant illumination settings such as the Middlebury dataset. Our best result is at 0.5 pixel accuracy in which it gets 14% bad pixels, which as of the late 2011 is in the top fourth of reported methods. At 1 pixel accuracy we get 8.3% bad pixels, which is on the top half of reported methods. In both cases we're worse than (but near) 2op.

6.5 Conclusion

Our paper makes two contributions. First, we derive a new MRF for stereo matching that encodes surface orientation. A unary cost allows us to adapt window shapes to account for foreshortening, while we also develop a metric pairwise cost that favors smooth surfaces. Second, we show experimentally that accounting for slant and tilt can improve performance in situations with wide baselines and lighting

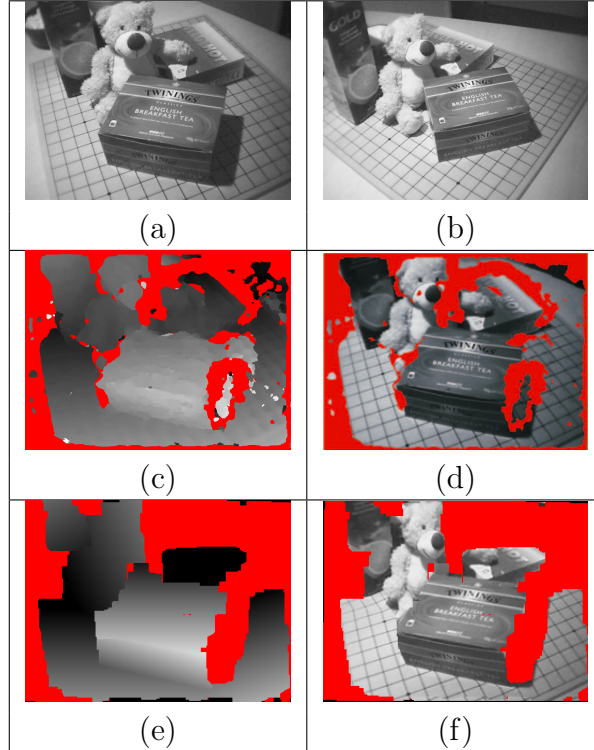


Figure 6.12: Evaluation of our method under slanted surfaces and a large baseline. Top row: (a) and (b) are the left and right image. (c) and (d) disparity map and image warpings for DAISY (reproduced from [76]) warped images built using the disparity map. Red pixels are occluded. (c) and (d) the disparity map and image warpings for our Slant+Tilt method. In the bottom row, our methods appear to show a slightly different viewpoint due to rectification.

variation. We do this primarily by comparing algorithms that are identical except for these novel features. Our formulation can be applied to other window based image comparison methods for stereo.

Chapter 7

Trainable 3D Recognition Using Stereo Matching

In previous chapters we have used Stereo matching for face recognition in the presence of pose variation. In this approach, stereo matching is used to compare two 2-D images based on correspondences that reflect the effects of viewpoint variation and allow for occlusion. We now show how to use stereo matching to derive image descriptors that can be used to train a classifier. This improves face recognition performance, producing the best published results on the CMU PIE dataset. We also demonstrate that classification based on stereo matching can be used for general object classification in the presence of pose variation. In preliminary experiments we show promising results on the 3D object class dataset, a standard, challenging 3D classification data set.

7.1 Introduction

In this chapter we make two contributions. First, one limitation of the work presented in Chapter 3 is that the image comparison does not produce a set of

descriptors that can be used to train a classifier. We show how to extract descriptors from stereo matching for use in classification, and show that this produces significantly more accurate face recognition in a standard data set. Second, we run a proof-of-concept experiment that shows that stereo matching can potentially be used to build classifiers for non-face objects. While preliminary, these experiments show that stereo matching may contribute more generally to 3D object recognition.

Our approach is simple:

1. Given two images we compute the epipolar geometry and rectify the pair of images,
2. We then use stereo matching to compute a descriptor. This descriptor encodes a matching cost for each pixel in each of the two images.
3. This descriptor is used to train a Support Vector Machine (SVM). At classification time, we apply the SVM to this descriptor and use the output of the SVM as a measure of similarity of the two images.

Figure 7.1 shows a diagram of our approach.

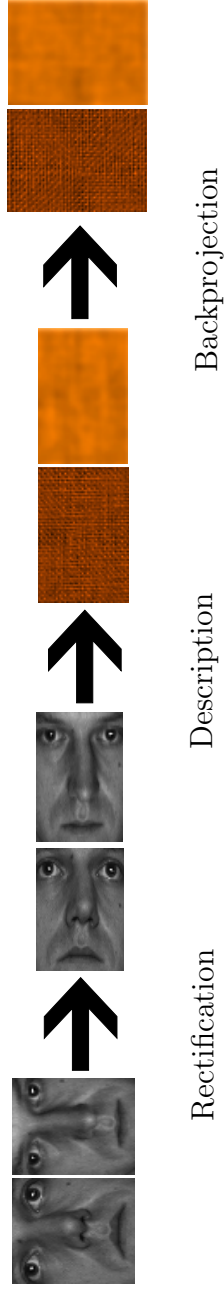


Figure 7.1: Diagram of our approach. In the rectification step, the images are rectified according to the epipolar geometry. In the description step the descriptors of pixelwise costs are computed. In the backprojection step, the pixelwise costs are transformed back to their original position, by applying the inverse of the rectification transform.

We describe two sets of experiments. First, we train a stereo-based comparison method for face recognition using the CMU PIE dataset[70]. In this case, training requires knowledge of the face pose; this is a reasonable assumption, since commercial, off-the-shelf systems exist for determining pose (eg., [57]). In this domain, we show that training can eliminate almost half the errors of a stereo-based face recognition system. Our results significantly exceed all published results in accuracy.

Second, as a proof of concept we evaluate on an object class detection experiment on an existing, challenging dataset[66]. Our results are preliminary, in that we do make use of a mask of the object of interest in determining the epipolar geometry relating two images. However, we then compare images using stereo matching and no object mask, and obtain very encouraging results.

The rest of the Chapter is organized as follows. Section 7.2 discusses related work. Section 7.3 describes the computation of the stereo matching costs, the image representation, the computation of the descriptors and the epipolar geometry. Section 7.4 presents and analyzes all experiments. Section 7.5 concludes.

7.2 Related Work

While there have been many recent advances in object categorization when there are limited viewpoint change [6, 26, 79, 25], there is also a significant body of work on the topic of object recognition using 3D representation and recognition. One important work is the 3D part based model of Savarese and Li [66]. Instead of recovering full 3D geometry these parts are connected using their mutual homographic transformation. Also, as part of this work, the authors collected a large and very hard dataset for 3D object class detection, which we use to evaluate our method.

The work of Thomas, et al. [75] presents ideas related to ours, in that they emphasize dense correspondences. However our approach is much simpler and seems to obtain better results.

Additionally, several works emphasize obtaining “trusted” correspondences and growing them such the work of Kushal and Ponce [43] and the work of Ferrari, et al. [27].

In our work we perform recognition by training a two-class classifier to distinguish between stereo-based descriptors from the same or different classes. This approach has been previously used by Moghaddam et al. [53] and Phillips [60], where the authors use the description of face differences for recognition purposes.

7.3 Stereo Matching Pixel-wise Descriptors

We build on our prior work [14], which makes use of the stereo algorithm of Criminisi et al. [22] to compute a distance between images. Using this stereo algorithm, we extract descriptors that can be used for classification.

7.3.1 Stereo Matching

The stereo algorithm uses dynamic programming (DP) to find the minimum cost matching between two corresponding scanlines. The important observation is that each step in the solution accounts for a single pixel in one of the two images. This is done using four planes (or cost matrices) called C_{Lo} , C_{Lm} , C_{Ro} and C_{Rm} . Each point in a matrix represents the last point in each image that has been accounted for, along with the nature of the last step used to account for a point. Points are accounted for by matching (m) and occlusions (o) in the left (L) and right

(R) images. The planes allow one to benefit or bias against state continuity. For example, beginning an occluded region may cost more than continuing an occlusion.

The algorithm is described in detail in Chapter 3.

7.3.2 Descriptor Generation

For recognition purposes, it is important that different areas of the image be given different weight when making the final same/not-same decision, similar to what is advocated in [53].

We are interested in describing the image differences between two images when there is change in viewpoint. To do so we build a descriptor using stereo matching. Each image is a sequence of scanlines. Each scanline has a corresponding scanline on the other image. Given two scanlines s_1 and s_2 of length l_1 and l_2 respectively the stereo method computes an optimal matching R which is a sequence (a word) of length $l_1 + l_2$. The optimal matching will be a sequence of symbols in the alphabet: $\Sigma = \{C_{Lo}, C_{Lm}, C_{Ro}, C_{Rm}\}$. Each symbol accounts for one pixel in either s_1 or s_2 .

This means that we can associate a cost with each pixel in each image being compared. These costs are the descriptor. The costs are calculated by computing the optimal matching R and then applying the rules presented in Table 7.1. Intuitively, Table 7.1 shows how to decode a matching. All possible combinations of two letters are shown along with the formula to decode this.

For each pair of scanlines s_1 and s_2 , this will generate the descriptor D_1 (which has as many costs as the length of s_1) and the descriptor D_2 (which has as many costs as the length of s_2). Location $D_{1,i}$ indicates how good a match was found s_2 for the pixel in $s_{1,i}$, and similarly location $D_{2,i}$ indicates how good a match was found s_1 for the pixel in $s_{2,i}$.

Table 7.1: Decoding of a matching $W = \langle c_1, \dots, c_n \rangle$ into two descriptors D_1 and D_2 of the same length of the scanlines matched.

c_k	c_{k-1}	$D_{1,i}$	$D_{2,j}$
C_{Lo}	C_{Lo}	α	-
C_{Lo}	C_{Lm}	β	-
C_{Lo}	C_{Rm}	β	-
C_{Ro}	C_{Lm}	-	β
C_{Ro}	C_{Ro}	-	α
C_{Ro}	C_{Rm}	-	β
C_{Lm}	C_{Lo}	$\beta' + M(i, j)$	-
C_{Lm}	C_{Lm}	$\gamma + M(i, j)$	-
C_{Lm}	C_{Ro}	$\beta' + M(i, j)$	-
C_{Lm}	C_{Rm}	$M(i, j)$	-
C_{Rm}	C_{Lo}	-	$\beta' + M(i, j)$
C_{Rm}	C_{Lm}	-	$M(i, j)$
C_{Rm}	C_{Ro}	-	$\beta' + M(i, j)$
C_{Rm}	C_{Rm}	-	$\gamma + M(i, j)$

7.3.3 Image Representation

The image representation is encoded into our method in the data used for $M(l, r)$ in Eqn. 3.8. We have evaluated describing images using the SIFT-like [47] DHOG descriptor from the VLFeat library [78] and windowed normalized sum of square differences (NSSD). We have found that when the intra-class variation is large, the DHOG descriptor is more accurate than the windowed NSSD image comparison metric, but when the intra-class variation is small, windowed NSSD performs much better than DHOG.

The usage of different image representations is very important feature of our method as it allows the method to be retargetted to different types of recognition tasks as will be shown in the experimental evaluation section.

7.3.4 Epipolar Geometry

One key step of our method is the aligning of the features before obtaining stereo correspondences. This can be done in one of two ways:

1. We can use a small number of feature points to compute the epipolar geometry of simple uniform objects such as faces in face recognition experiments. In our experiments we use 4 hand-clicked points to compute a scaled-orthographic epipolar geometry, as described in [14]. These feature points, in the case of faces, can be automatically found with good accuracy, for example using a commercial system such as Omron [57].
2. We can use off-the-shelf methods like the one of Domke and Aloimonos [24] to automatically compute the epipolar geometry of less uniform objects.

7.3.5 Classification

At training time, once the descriptors have been generated we train a linear SVM classifier on the same/not same task, similar to [60, 53]¹. At test time, when a new pair of images needs to be compared we compute the descriptor and apply the SVM. We use the SVM signed distance to the margin as a measure of the similarity between the two images.

We use LIBSVM [18] and use cross validation to set the value of C . As $M(l, r)$ is usually bounded from above and below, for example in the case of NSSD by 0 and 1, it is not necessary to perform normalization of the classifiers. In the cases in which $M(l, r)$ is not bounded from above and below, we linearly scale the values in the descriptors to the range $[0, 1]$.

¹We also tried using a Gaussian RBF kernel and the results were comparable. We adopted a linear kernel since it has the benefit simplicity.

For effective classification, after rectification, description and backprojection, the features should be aligned between different image pairs. This happens in the case of faces where in each pose the features are aligned with a similarity transformation to begin with. With general 3D objects this happens to a lesser degree, because unlike with faces knowing the pose of a 3D object is nearly as hard as knowing its class. Still, in the case of 3D objects the classifier is able to learn useful things from the descriptor, as our experiments will show.

7.4 Experimental Evaluation

We evaluate our method in two experiments showing, in both cases, state-of-the-art results:

1. We evaluate on the CMU PIE dataset on a gallery-probe face recognition experiment across pose.
2. We evaluate on the 3D Object Categories Dataset of Savarese and Li. in an object categorization task. Note that in this experiment, while we achieve strong results, we have not yet fully explored the problem of determining the epipolar geometry.

7.4.1 CMU PIE Experiments

We perform a gallery-probe recognition experiment on the CMU PIE dataset. In this experiment we use half the individuals to train and half the individuals to test. For each pair of poses we perform a gallery-probe experiment with all the individuals in the testing set. The Figure 7.3 shows images from all poses in the CMU PIE dataset. The results for this experiment are shown in Table 7.2.

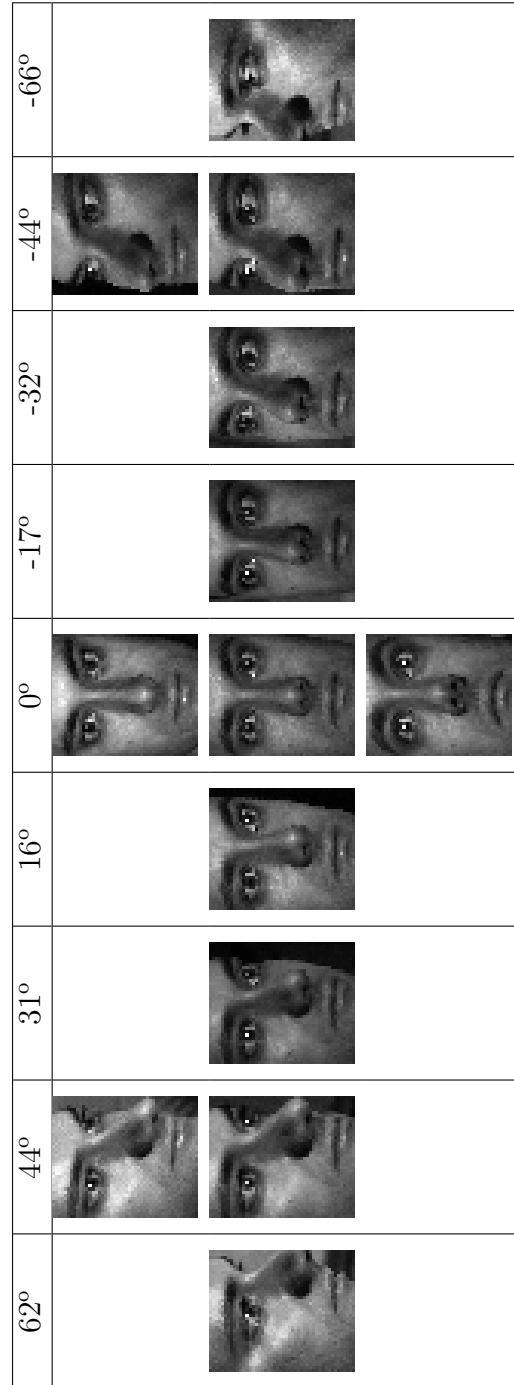


Figure 7.2: Images from the 13 poses in the CMU PIE dataset.

In these experiments we used NSSD based descriptors, that is, the value of $M(l, r)$ in Eqn. 3.8 is based on NSSD.

Our new method is identical to the Stereo Matching Distance (SMD) of [14] except that we derive descriptors from stereo matching and use them to train an SVM, while SMD simply sums the stereo matching cost and uses it as a measure of similarity. Our new approach requires knowledge of the face pose, so that an appropriate SVM may be trained. This is reasonable for faces, since commercial systems such as [57] exist for pose determination. With this added training, our system eliminates almost half the errors produced by SMD. It also creates a large improvement over the best currently published results.

The protocol is the following:

1. Given a gallery pose and probe pose. Train a SVM using half of all the CMU PIE data for that gallery-probe, that is: 34 positive descriptors and $34 * 33$ negative descriptors.
2. Test on the other half of CMU PIE, by performing 34 independent gallery probe experiments, on the other half of the data. Build a gallery with 34 individuals, and evaluate the accuracy on the 34 probes. Given the cost of a

Table 7.2: A comparison of recognition accuracy averaged across pose for our descriptor-based stereo matching distance with other methods on CMU PIE.

34 Faces - CMU PIE

Method	Accuracy
Eigen light-fields (Multi-point norm.) [33]	66.3%
SMD (Castillo and Jacobs [14])	86.8%
Slant SMD [15]	90.1%
Partial Least Squares [69]	90.1%
Descriptor SMD	93.2%

probe to 34 gallery images, we determine the identity of the probe to be the identity of the gallery with the largest signed distance to the margin.

There are other methods that evaluate on the CMU PIE dataset but use the full 68 images (instead of 34) to test either because they don't require training (such as SMD and Slant SMD) or because they build their models on other images (such as 3D morphable models-based method LiST [63].) However, on 68 individuals SMD outperforms LiST, and on 34 individuals Descriptor SMD clearly outperforms SMD, so it would be expected that Descriptor SMD outperforms LiST.

7.4.2 3D Object Categories Dataset Experiments

In these experiments we evaluate our stereo matching method on a hard object categorization task. Our main goal is to show that stereo matching has the potential to be useful for classification tasks, in which variation in appearance is due, not only to changes in pose, but also to within-class variations in shape and appearance.



Figure 7.3: One example from each class of the 3D Object Category Dataset. Top row: bicycle, car, cellphone and iron. Bottom row: mouse, shoe, stapler and toaster

We tried to faithfully replicate the experimental setup described in the object categorization experiments in [66]. We randomly selected 200 images of 7 object instances to train and 70 randomly selected testing images of 4 novel object instances. The classification task was on the same 8 categories as Savarese and Li, namely: stapler, bicycle, car, cellphone, iron, mouse, shoe and toaster. The results of this experiment are shown in Table 7.3.

Determining epipolar geometry for images in this data set poses a significant challenge. Two images may contain objects from the same class, set in completely different backgrounds. In face recognition, epipolar geometry is computed after face detection, so that the computation can be based on corresponding objects. In the data set of [66], it is not appropriate to compute epipolar geometry based on the entire image, since the two scenes do not correspond.

For our proof-of-concept experiments we have used the image masks to focus on the parts of the image containing the object, and used the algorithm of Domke and Aloimonos [24] to then automatically compute the epipolar geometry based on these regions of interest. We stress that masks are not used to crop the images for stereo matching, only to compute epipolar geometry; matching is performed over the entire image.

Clearly this approach will not produce a meaningful epipolar geometry when applied two image from two different classes, with quite different appearances. This is not a problem, since these will lead to high costs, indicating that the objects do not match well. However, even determining an epipolar geometry between objects of the same class can be a significant problem, since there can be a lot of shape variation. There are several ways in which we can attempt to overcome this limitation. One way is to use a RANSAC-like approach to compute the top k candidate epipolar

geometries, and then perform stereo matching using these k rectifications, returning the one that gives the lowest-cost result. This direction is left as future work.

For the objects: bicycle, iron, shoe, toaster and car, we obtain better accuracies than Savarese and Li, for the objects: cellphone, mouse and stapler we obtain accuracies that are worse than the results of Savarese and Li. Overall we obtain a global accuracy of 80% while Savarese and Li obtain an accuracy of 75.7%. See Table 7.3 for the complete results. In these experiments we used DHOG based descriptors, that is, the value of $M(l, r)$ in Eqn. 3.8 is based on DHOG, computed using [78].

The protocol is the following:

1. Given 200 training images of objects and 70 testing images of novel objects, compute $(200 * 199)/2 = 19990$ descriptors, and train a SVM with labels depending if each descriptor are of the same class of object or not.
2. For each test image compute the descriptor of the each training image. Query the SVM and declare the class of the test image to be the class of the training image that had the highest signed distance to margin.

We feel that these results are intriguing. Clearly, stereo matching is not appropriate for some objects, in which different instances of the same class have very different shapes, so that corresponding points cannot be related by matching along epipolar lines. However, our results indicate that for some objects, these shape variations may be relatively small compared to the effects of viewpoint variation. At the same time, our experiments indicate that stereo matching is robust to effects of clutter, in which the object of interest has not been fully separated from the background. This suggests that stereo matching may be an effective tool for classification of 3D objects.

Table 7.3: Confusion matrix for the 3D object category detection experiment. The overall accuracy over the 70 test objects is 80%.

	cellphone	bicycle	iron	mouse	shoe	stapler	toaster	car
cellphone	0.60				0.20			0.20
bicycle		1.00						
iron			0.88				0.12	
mouse				0.43	0.21	0.14		0.21
shoe					0.75			0.25
stapler	0.25					0.75		
toaster		0.09					0.91	
car								1.00

7.5 Conclusion

There have been great strides in developing image descriptors for 2-D, appearance based recognition. When we seek to identify 3D objects from arbitrary viewpoints, the way in which we match image descriptors should be informed by the geometric constraints induced by changes in pose. Stereo matching makes use of these constraints, while also allowing for occlusions caused by changes in viewpoint. In order to implement image comparisons that are informed by knowledge of 3D, we have presented a simple method for recognition based on stereo matching. The same method obtains state-of-the-art results on two very different recognition tasks. Our results in face recognition significantly exceed those of prior approaches. Our results on more general classification tasks are preliminary, but demonstrate the relevance of stereo matching to these tasks as well.

Chapter 8

Conclusion

The main insight of this dissertation is that stereo matching can be used for face recognition. This finding led us to develop new stereo algorithms that improve face recognition and that are interesting in their own right.

8.1 Stereo Matching for Face Recognition Across Pose

The core of this dissertation is a new, effective method for face recognition across pose that is decidedly 2D, fast and practical.

Correspondences are fundamental to recognize faces across pose. We want to compare images from different viewpoints by finding correspondences. The methods we developed use stereo matching to measure the similarity of two images. Stereo matching is a standard problem in computer vision in which correspondences are obtained between two images of the same scene. The typical application of stereo matching is as a method to reconstruct a 3D model of a scene. We don't perform reconstruction, rather we use the stereo matching cost (the cost optimized in the

process of finding correspondences) as a measure of similarity of two faces. Stereo matching is a well understood problem and provides a firm foundation to build on.

We have built a fast, practical method for stereo matching in the presence of medium pose variation [14]. Also, we have built a method that is robust to large and very large changes in viewpoint and illumination when matching very slanted objects [15]. We have evaluated this approach both in controlled settings (like PIE) with both methods outperforming all prior work. Our results are almost perfect for horizontal pose differences of up to 30 degrees, and beyond that the results gracefully degrade.

8.2 Dense Wide-baseline Matching with Varying Illumination

Face recognition is an application of stereo matching where the illumination can vary significantly. Using our insights from stereo matching to compare faces, we have developed a formulation that allows us to adapt a 2-D Markov Random Field based stereo formulation for wide baseline dense matching with variation in illumination.

Illumination change is almost always handled by normalizing the comparison function inside a relatively large window. The size of these windows is affected by foreshortening. If we do not account for this effect, we incur misalignments that are systematic and significant and are exacerbated by wide baseline conditions.

We have developed a general formulation of dense wide baseline stereo with varying illumination that adjusts the size of the matching window and designed two methods to solve them. The general formulation includes a novel smoothness term

that encodes surface orientation. This smoothness term has properties that make it amenable to optimization. The first method is based on dynamic programming and fully accounts for the effect of slant. The second method is based on graph cuts and fully accounts for the effect of slant and tilt, presented in Chapter 6. Our results show that this energy function is very robust to changes in illumination that occur in wide baseline stereo and our results compare favorably with other methods to handle wide baseline dense stereo matching.

8.3 Descriptor-based Learning for Face Verification

One inherent limitation of the stereo-based methods we have developed is that they weigh each location in the face equally. There are strong reasons to believe that this is not a good idea; differences around the eyes should be more significant than differences in the cheeks, because there are areas of the face that are more strongly connected with identity like the area around the eyes and there areas that are not strongly connected with identity like the cheeks. Ideally we should give areas that are strongly connected with identity more weight than areas that are weakly connected with identity. We have developed methods to integrate learning into our stereo-based face recognition work. In this method we can use available data to learn how to weigh each pixel differently in the process of determining the image similarity. This formulation will allow us to learn how to compensate for slight variations in the images that are not being explicitly accounted for by the (pose+illumination) model described in the previous section. These variations include: expression changes, aging, weight variation, etc. We have evaluated this

approach both in controlled settings (like PIE) and in unconstrained settings using data sets like Labeled Faces in the Wild, and the results are encouraging. When we did this we obtained better results than all published results on the widely used Labeled Faces in the Wild (LFW) dataset. To learn more about this method, see [\[16\]](#).

8.4 Onwards

By using dense wide-baseline stereo we have made progress in face recognition across pose, but the problem is definitely not solved. While obtaining correspondences is fundamental, and our proposed stereo methods are quite good at doing so, there are many other issues that make face recognition across pose very hard: choice of representation/image comparison, mixture of rigid and non-rigid deformations, interactions with expression and illumination, camera motion and blur, ephemeral imaging conditions (for example: eyes open/closed, mouth open/close, teeth visible).

There are several ways in which our work could be extended in the future to handle such difficulties. These extensions will require significant amounts of work, but are feasible. For example we could handle pose and expression by computing dense correspondences with stereo + optical flow. In this case we would proceed as in stereo finding correspondences along epipolar lines but the algorithm would also be allowed to look for correspondences across epipolar lines, in this case paying some type of penalty.

Bibliography

- [1] <http://www.uni-bonn.de/~uzs751/MRTStereo/>. 101, 103
- [2] Yael Adini, Yael Moses, and Shimon Ullman. Face recognition: The problem of compensating for changes in illumination direction. *IEEE Trans. Pattern Anal. Mach. Intell.*, 19(7):721–732, 1997. 15
- [3] Ahmed Bilal Ashraf, Simon Lucey, and Tsuhan Chen. Learning patch correspondences for improved viewpoint invariant face recognition. In *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2008. 14
- [4] Ronen Basri and David Jacobs. Lambertian reflectance and linear subspaces. *IEEE TPAMI*, 25(2):218–233, 2003. 12
- [5] Peter N. Belhumeur. A Bayesian approach to binocular stereopsis. *IJCV*, 19(3):237–260, 1996. 71, 72, 91
- [6] Alexander C. Berg, Tamara L. Berg, and Jitendra Malik. Shape matching and object recognition using low distortion correspondences. In *CVPR (1)*, pages 26–33. IEEE Computer Society, 2005. 113
- [7] David Beymer and Tomaso Poggio. Face recognition from one example view. Technical Report AIM-1536, , 1995. 11

- [8] Stan Birchfield and Carlo Tomasi. A pixel dissimilarity measure that is insensitive to image sampling. *IEEE TPAMI*, 20(4):401–406, 1998. [20](#), [102](#)
- [9] Volker Blanz and Thomas Vetter. Face recognition based on fitting a 3d morphable model. *IEEE Trans. Pattern Anal. Mach. Intell.*, 25(9):1063–1074, 2003. [12](#), [53](#)
- [10] Michael Bleyer, Carsten Rother, and Pushmeet Kohli. Surface stereo with soft segmentation. In *CVPR*, pages 1570–1577. IEEE, 2010. [22](#), [88](#), [100](#)
- [11] Kevin W. Bowyer, Kyong I. Chang, and Patrick J. Flynn. A survey of approaches and challenges in 3d and multi-modal 3d + 2d face recognition. *Computer Vision and Image Understanding*, 101(1):1–15, 2006. [14](#)
- [12] Yuri Boykov, Olga Veksler, and Ramin Zabih. Fast approximate energy minimization via graph cuts. *IEEE TPAMI*, 23(11):1222–1239, 2001. [21](#), [87](#)
- [13] Carlos D. Castillo and David W. Jacobs. Using stereo matching for 2-d face recognition across pose. In *CVPR*, 2007. [60](#)
- [14] Carlos D. Castillo and David W. Jacobs. Using stereo matching with general epipolar geometry for 2d face recognition across pose. *IEEE Trans. Pattern Anal. Mach. Intell.*, 31(12):2298–2304, 2009. [viii](#), [ix](#), [76](#), [77](#), [78](#), [81](#), [82](#), [83](#), [84](#), [114](#), [117](#), [120](#), [127](#)
- [15] Carlos D. Castillo and David W. Jacobs. Face recognition with large pose variation. In *CVPR*, 2011. [22](#), [85](#), [87](#), [120](#), [127](#)
- [16] Carlos D. Castillo and David W. Jacobs. Trainable 3d recognition using stereo matching. In *3D Representation and Recognition Workshop at ICCV*, 2011. [129](#)

- [17] Xiujuan Chai, Shiguang Shan, Xilin Chen, and Wen Gao. Locally linear regression for pose-invariant face recognition. *IEEE Transactions on Image Processing*, 16(7):1716–1725, 2007. [viii](#), [2](#), [5](#), [13](#), [43](#), [48](#), [49](#), [57](#), [77](#)
- [18] Chih-Chung Chang and Chih-Jen Lin. *LIBSVM: a library for support vector machines*, 2001. Software available at <http://www.csie.ntu.edu.tw/~cjlin/libsvm>. [60](#), [117](#)
- [19] C.W. Chen and C.L. Huang. Human facial feature extraction for face interpretation and recognition. pages II:204–207, 1992. [30](#)
- [20] Hansen F. Chen, Peter N. Belhumeur, and David W. Jacobs. In search of illumination invariants. In *CVPR*, pages 1254–1261, 2000. [16](#)
- [21] I. J. Cox, S. L. Hingorani, S. B. Rao, and B. M. Maggs. A maximum likelihood stereo algorithm. *Computer Vision and Image Understanding*, 63(3):542–567, 1996. [36](#), [37](#), [71](#)
- [22] Antonio Criminisi, Andrew Blake, Carsten Rother, Jamie Shotton, and Philip H. S. Torr. Efficient dense stereo with occlusions for new view-synthesis by four-state dynamic programming. *International Journal of Computer Vision*, 71(1):89–110, 2007. [19](#), [36](#), [38](#), [39](#), [53](#), [71](#), [74](#), [87](#), [114](#)
- [23] F. Devernay and O. Faugeras. Computing differential properties of 3-d shapes from stereoscopic images without 3-d models. In *CVPR*, 1994. [19](#), [20](#), [21](#), [87](#)
- [24] Justin Domke and Yiannis Aloimonos. A probabilistic notion of correspondence and the epipolar constraint. In *3DPVT*, pages 41–48. IEEE Computer Society, 2006. [117](#), [123](#)

- [25] Pedro F. Felzenszwalb, Ross B. Girshick, David A. McAllester, and Deva Ramanan. Object detection with discriminatively trained part-based models. *IEEE Trans. Pattern Anal. Mach. Intell.*, 32(9):1627–1645, 2010. **113**
- [26] Robert Fergus, Pietro Perona, and Andrew Zisserman. Object class recognition by unsupervised scale-invariant learning. In *CVPR (2)*, pages 264–271. IEEE Computer Society, 2003. **113**
- [27] Vittorio Ferrari, Tinne Tuytelaars, and Luc J. Van Gool. Simultaneous object recognition and segmentation from single or multiple model views. *International Journal of Computer Vision*, 67(2):159–188, 2006. **114**
- [28] Andrea Fusiello and Luca Irsara. Quasi-euclidean uncalibrated epipolar rectification. In *ICPR*, pages 1–4. IEEE, 2008. **104**
- [29] A.S. Georghiades, P.N. Belhumeur, and D.J. Kriegman. From few to many: Illumination cone models for face recognition under variable lighting and pose. *IEEE Trans. Pattern Anal. Mach. Intelligence*, 23(6):643–660, 2001. **12**
- [30] Y. Gizatdinova and V. Surakka. Feature-based detection of facial landmarks from neutral and expressive facial images. 28(1):135–139, January 2006. **30**
- [31] Raghuraman Gopalan and David Jacobs. Comparing and combining lighting insensitive approaches for face recognition. *Computer Vision and Image Understanding*, 114(1):135 – 145, 2010. **16, 19**
- [32] W. Eric Grimson. *From Images To Surfaces: A Computational Study of the Human Early Vision System*. MIT Press, 1981. **91**
- [33] Ralph Gross, Simon Baker, Iain Matthews, and Takeo Kanade. Face recognition across pose and illumination. In Stan Z. Li and Anil K. Jain, editors, *Handbook*

- of Face Recognition*. Springer-Verlag, June 2004. 2, 5, 13, 43, 48, 52, 77, 78, 120
- [34] Ralph Gross and Vladimir Brajovic. An image preprocessing algorithm for illumination invariant face recognition. In *4th International Conference on Audio- and Video-Based Biometric Person Authentication (AVBPA)*. Springer, June 2003. 52
- [35] Ralph Gross, Iain Matthews, and Simon Baker. Appearance-based face recognition and light-fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(4):449 – 465, April 2004. viii, 2, 5, 43, 48, 49, 57
- [36] Ralph Gross, Jianbo Shi, and Jeffrey Cohn. Quo vadis face recognition? In *Third Workshop on Empirical Evaluation Methods in Computer Vision*, December 2001. viii, 45, 46, 50, 51, 81
- [37] Matthieu Guillaumin, Jakob J. Verbeek, and Cordelia Schmid. Is that you? metric learning approaches for face identification. In *ICCV*, pages 498–505, 2009. 18
- [38] S.M. Hanif, L. Prevost, R. Belaroussi, and M. Milgram. Real-time facial feature localization by combining space displacement neural networks. 29(8):1094–1104, June 2008. 30
- [39] Gary B. Huang, Manu Ramesh, Tamara Berg, and Erik Learned-Miller. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. In *Faces in Real-Life Images Workshop in European Conference on Computer Vision (ECCV)*, 2008. 17

- [40] O. Jesorsky, K. Kirchberg, and R. Frischolz. Robust face detection using the hausdorff distance. *Audio and Video Based Person Authentication*, pages 90–95, 2001. [17](#)
- [41] Andreas Klaus, Mario Sormann, and Konrad F. Karner. Segment-based stereo matching using belief propagation and a self-adapting dissimilarity measure. In *ICPR*, pages 15–18, 2006. [21](#)
- [42] Neeraj Kumar, Alexander C. Berg, Peter N. Belhumeur, and Shree K. Nayar. Attribute and simile classifiers for face verification. In *ICCV*, pages 365–372, 2009. [17](#)
- [43] Akash Kushal and Jean Ponce. Modeling 3d objects from stereo views and recognizing them in photographs. In Ales Leonardis, Horst Bischof, and Axel Pinz, editors, *ECCV (2)*, volume 3952 of *Lecture Notes in Computer Science*, pages 563–574. Springer, 2006. [114](#)
- [44] Gang Li and Steven W. Zucker. Stereo for slanted surfaces: First order disparities and normal consistency. In *EMMCVPR*, pages 617–632, 2005. [19](#), [20](#), [21](#), [69](#), [87](#), [89](#)
- [45] Gang Li and Steven W. Zucker. Differential geometric inference in surface stereo. *IEEE TPAMI*, 32(1):72–86, 2010. [21](#), [87](#), [104](#)
- [46] H. Ling, S. Soatto, N. Ramanathan, and D. W. Jacobs. A study of face recognition as people age. In *International Conference on Computer Vision (ICCV)*, 2007. [17](#), [60](#)
- [47] David G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004. [87](#), [91](#), [116](#)

- [48] Simon Lucey and Tsuhan Chen. A viewpoint invariant, sparsely registered, patch based, face verifier. *International Journal of Computer Vision (IJCV)*, December 2007. [13](#)
- [49] Jiri Matas, Ondrej Chum, Martin Urban, and Tomás Pajdla. Robust wide baseline stereo from maximally stable extremal regions. In Paul L. Rosin and A. David Marshall, editors, *BMVC*. British Machine Vision Association, 2002. [6](#), [19](#), [67](#)
- [50] Quinn McNemar. Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika*, 12:153–157, 1947. [80](#)
- [51] Peter Meer and Bogdan Georgescu. Edge detection with embedded confidence. *IEEE TPAMI*, 2001. [100](#)
- [52] Krystian Mikolajczyk and Cordelia Schmid. A performance evaluation of local descriptors. *IEEE Trans. Pattern Anal. Mach. Intell.*, 27(10):1615–1630, 2005. [91](#)
- [53] Baback Moghaddam, Tony Jebara, and Alex Pentland. Bayesian modeling of facial similarity. In *Proceedings of the 1998 conference on Advances in neural information processing systems II*, pages 910–916, Cambridge, MA, USA, 1999. MIT Press. [17](#), [114](#), [115](#), [117](#)
- [54] Eric Nowak and Frédéric Jurie. Learning visual similarity measures for comparing never seen objects. In *CVPR*, 2007. [60](#), [64](#)
- [55] Abhijit S. Ogale and Yiannis Aloimonos. Stereo correspondence with slanted surfaces: Critical implications of horizontal slant. In *CVPR (1)*, pages 568–573, 2004. [21](#)

- [56] Abhijit S. Ogale and Yiannis Aloimonos. Shape and the stereo correspondence problem. *IJCV*, 65(3):147–162, 2005. 21, 87, 89
- [57] Omron. OKAO vision. http://www.omron.com/r_d/coretech/vision/okao.html, 2009. 113, 117, 120
- [58] Margarita Osadchy, David W. Jacobs, and Michael Lindenbaum. On the equivalence of common approaches to lighting insensitive recognition. In *ICCV*, pages 1721–1726. IEEE Computer Society, 2005. 16
- [59] P. J. Phillips, P. J. Flynn, T. Scruggs, K. W. Bowyer, and W. Worek. Preliminary face recognition grand challenge results. *FGR: International Conference on Automatic Face and Gesture Recognition, Proceedings of*, pages 15–24, 2006. 18
- [60] P. Jonathon Phillips. Support vector machines applied to face recognition. In *Advances in Neural Information Processing Systems 11*, pages 803–809. MIT Press, 1998. 60, 114, 117
- [61] Simon J. D. Prince, James H. Elder, Jonathan Warrell, and Fatima M. Felisberti. Tied factor analysis for face recognition across large pose differences. *IEEE Trans. Pattern Anal. Mach. Intell.*, 30(6):970–984, 2008. 14
- [62] Narayanan Ramanathan and Rama Chellappa. Face verification across age progression. *IEEE Transactions on Image Processing*, 15(11):3349–3362, 2006. 17
- [63] Sami Romdhani, Volker Blanz, and Thomas Vetter. Face identification by fitting a 3d morphable model using linear shape and texture error functions. In *Computer Vision – ECCV’02*, volume 4, pages 3–19, Copenhagen, Denmark,

2002. [viii](#), [ix](#), [2](#), [5](#), [12](#), [43](#), [45](#), [46](#), [48](#), [49](#), [50](#), [51](#), [53](#), [55](#), [56](#), [57](#), [77](#), [78](#), [81](#), [82](#), [83](#), [121](#)
- [64] Carsten Rother, Vladimir Kolmogorov, Victor S. Lempitsky, and Martin Szummer. Optimizing binary mrfs via extended roof duality. In *CVPR*, 2007. [21](#), [87](#)
- [65] P. Sankaran, S. Gundimada, R.C. Tompkins, and V.K. Asari. Pose angle determination by face, eyes and nose localization. pages III: 161–161, 2005. [30](#)
- [66] Silvio Savarese and Fei-Fei Li. 3d generic object categorization, localization and pose estimation. In *ICCV*, pages 1–8, 2007. [113](#), [123](#)
- [67] Daniel Scharstein and Richard Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *International Journal of Computer Vision*, 47(1-3):7–42, 2002. [8](#), [87](#)
- [68] William Robson Schwartz, Huimin Guo, and Larry S. Davis. A robust and scalable approach to face identification. In Kostas Daniilidis, Petros Maragos, and Nikos Paragios, editors, *ECCV (6)*, volume 6316 of *Lecture Notes in Computer Science*, pages 476–489. Springer, 2010. [18](#)
- [69] Abhishek Sharma and David W. Jacobs. Bypassing synthesis: PLS for face recognition with pose, low-resolution and sketch. In *CVPR*, 2011. [14](#), [78](#), [120](#)
- [70] Terence Sim, Simon Baker, and Maan Bsat. The cmu pose, illumination, and expression database. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(12):1615 – 1618, December 2003. [8](#), [13](#), [18](#), [42](#), [76](#), [86](#), [101](#), [113](#)
- [71] Noah Snavely, Steven M. Seitz, and Richard Szeliski. Photo tourism: exploring photo collections in 3d. *ACM Trans. Graph.*, 25(3):835–846, 2006. [85](#)

- [72] Jian Sun, Nanning Zheng, and Heung-Yeung Shum. Stereo matching using belief propagation. *IEEE TPAMI*, 25(7):787–800, 2003. 21, 87
- [73] Yaniv Taigman, Lior Wolf, and Tal Hassner. Multiple one-shots for utilizing class label information. In *BMVC*. British Machine Vision Association, 2009. 18
- [74] Xiaoyang Tan and Bill Triggs. Enhanced local texture feature sets for face recognition under difficult lighting conditions. *IEEE Transactions on Image Processing*, 19(6):1635–1650, 2010. 16
- [75] Alexander Thomas, Vittorio Ferrari, Bastian Leibe, Tinne Tuytelaars, Bernt Schiele, and Luc J. Van Gool. Towards multi-view object class detection. In *CVPR (2)*, pages 1589–1596. IEEE Computer Society, 2006. 114
- [76] Engin Tola, Vincent Lepetit, and Pascal Fua. Daisy: An efficient dense descriptor applied to wide-baseline stereo. *IEEE TPAMI*, 32(5):815–830, 2010. xi, 8, 87, 91, 101, 108, 109
- [77] M. A. Turk and A. P. Pentland. Face recognition using eigenfaces. pages 586–591, 1991. 43
- [78] A. Vedaldi and B. Fulkerson. VLFeat: An open and portable library of computer vision algorithms. <http://www.vlfeat.org/>, 2008. 116, 124
- [79] Paul Viola and Michael Jones. Robust real-time object detection. *International Journal of Computer Vision*, 2002. 17, 113
- [80] Haitao Wang, Stan Z. Li, and Yangsheng Wang. Face recognition under varying lighting conditions using self quotient image. In *FGR*, pages 819–824. IEEE Computer Society, 2004. 16

- [81] Laurenz Wiskott, Jean-Marc Fellous, Norbert Krüger, and Christoph von der Malsburg. Face recognition by elastic bunch graph matching. In Gerald Sommer, Kostas Daniilidis, and Josef Pauli, editors, *Proc. 7th Intern. Conf. on Computer Analysis of Images and Patterns, CAIP'97, Kiel*, number 1296, pages 456–463, Heidelberg, 1997. Springer-Verlag. [11](#)
- [82] Lior Wolf, Tal Hassner, and Yaniv Taigman. Descriptor based methods in the wild. In *Faces in Real-Life Images Workshop in European Conference on Computer Vision (ECCV)*, 2008. [18](#), [60](#)
- [83] Oliver J. Woodford, Philip H. S. Torr, Ian D. Reid, and Andrew W. Fitzgibbon. Global stereo reconstruction under second order smoothness priors. In *CVPR*, 2008. [88](#), [102](#), [104](#)
- [84] W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld. Face recognition: A literature survey. *ACM Comput. Surv.*, 35(4):399–458, December 2003. [2](#), [10](#)
- [85] Zihan Zhou, Arvind Ganesh, John Wright, Shen-Fu Tsai, and Yi Ma. Nearest-subspace patch matching for face recognition under varying pose and illumination. In *FG*, pages 1–8. IEEE, 2008. [77](#)