Wide-Baseline Stereo for Face Recognition with Large Pose Variation

Carlos D. Castillo and David W. Jacobs
Computer Science Department
University of Maryland, College Park
{carlos,djacobs}@cs.umd.edu

Abstract

2-D face recognition in the presence of large pose variations presents a significant challenge. When comparing a frontal image of a face to a near profile image, one must cope with large occlusions, non-linear correspondences, and significant changes in appearance due to viewpoint. Stereo matching has been used to handle these problems, but performance of this approach degrades with large pose changes. We show that some of this difficulty is due to the effect that foreshortening of slanted surfaces has on window-based matching methods, which are needed to provide robustness to lighting change. We address this problem by designing a new, dynamic programming stereo algorithm that accounts for surface slant. We show that on the CMU PIE dataset this method results in significant improvements in recognition performance.

1. Introduction

Recent work [7] 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 [18], these approaches generally do not produce a cost based on dense correspondences that is appropriate for image comparison and face recognition.

In this paper 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 of [7], as well as other previous approaches to face recognition with pose variation.

2. Related Work

Our research builds on two streams of work. First, we will describe work on face recognition in the presence of pose variation. Next we will discuss issues that arise in wide-baseline stereo matching.

2.1. Face Recognition with Pose Variation

There has been a tremendous amount of work on face recognition in recent years (see [25] for a review). We will focus on work that handles variations in pose (see [24] for a recent review that focuses on pose).

Early work handled some pose variations by matching 2-D images in ways that allowed for deformations. Wiskott et al. [23] did this using Elastic Bunch Graph Matching, while Beymer and Poggio [4] used matching with optical flow.

3-D models have also been used to account for pose variation. For example, Basri and Jacobs [2] and Georghiades et al. [12] construct 3-D models using controlled sensing, and align these with 2-D for recognition. Morphable models ([6, 21]) have also been used for this purpose. These models allow 3-D reconstruction of a face from a single image, taken under uncontrolled viewing conditions. This allows 3-D information to be used in comparing two face images. This method has also produced strong results on pose variations taken from the PIE dataset.

Recent work has also matched patches of images to provide robustness to pose variation. Chai et al. [8] present a learning method to perform patch-based rectification. Lucey and Chen [17] and Ashraf et al. [1] also describe patch-based algorithms for face recognition.

Our paper builds directly on our prior stereo matching approach [7]. In this work, images of two faces are compared by running a stereo matching algorithm and using the cost of the matching as a measure of the similarity of the faces. This method works quite well for moderate pose changes, but performance degrades as the change in pose increases.

2.2. 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.[10], 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 [13]; 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. [18]).

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 1. This issue is elegantly described by [16, 11]. Following Li and Zucker [16] 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 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 [16] and Devernay and Faugeras [11] 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, [16] and [11] 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 [5] 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 an image 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 [20], 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.

3. Stereo Matching with Slant

Our recent work [7] 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.

A stereo algorithm for face recognition should have the following characteristics:

1. We require dense correspondences. Every pixel in each image should be accounted for by matching or occlusion.
2. We require effective handling of wide baselines.
3. We require matching images with illumination variation.

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 [16]. 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 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.

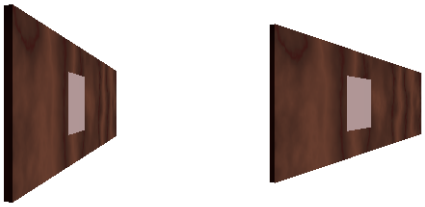

Figure 2. A wooden wall with a small patch marked seen from two viewpoints. This example illustrates the critical importance of handling slant correctly.

4. Dynamic Programming Algorithm

Dynamic programming (DP) approaches to stereo matching have been widely used [9, 10]. These are suitable for face recognition because they are fast and images of faces can be matched using a monotonicity constraint [3]. 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.

4.1. 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 [3].

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.

4.2. 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:

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

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.[10].

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

$$\frac{1}{2}\left[\frac{\sum_{\delta\in\Omega}\left((I^l_{p_l+\delta}-\overline{I^l_{p_l}})-(I^r_{p_r+\delta}-\overline{I^r_{p_r}})\right)^2}{\sum_{\delta\in\Omega}(I^l_{p_l+\delta}-\overline{I^l_{p_l}})^2+\sum_{\delta\in\Omega}(I^r_{p_r+\delta}-\overline{I^r_{p_r}})^2}\right] \quad (2)$$

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 3x7 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 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 $-1 < d' \le 3$ since other values signal an occlusion.

[1]We will abuse notation and define $\mathrm{NSSD}(l,r)$ 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.

4.3. 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:

$$C_{or}(x,d) = \min \begin{cases} \min_{\alpha} C_m(\alpha, x, d) + \gamma \\ \min_{d_p<d}(\lfloor d \rfloor - \lfloor d_p \rfloor)M + C_{or}(x, d_p) \end{cases} \tag{3}$$

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.

4.4. 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:

$$C_{ol}(x,d) = \min \begin{cases} \min_{\alpha} C_m(\alpha, x, d) + \gamma \\ \min_{k<x} Mk + C_{ol}(x-k, d+k) \end{cases} \tag{4}$$

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.

4.5. Total Cost for Recognition

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

$$T(x,d) = \min_{\beta} C_m(\beta, x, d) \tag{5}$$

The cost of matching between two stereo pairs is therefore $\min_d T(N, d)$. Following [7] 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. Experimental Evaluation

We have tested our algorithm using the CMU PIE dataset [22]. 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 matching 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) [7]. That method produces the best published results across the 13 pose conditions in the CMU PIE dataset. Also, since our algorithm is similar to [7] 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 1. We also compare with the method of Romdhani et al. [21] 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 [26, 14, 8]). Most of them don't evaluate using large variation in pose [26, 8] and for the ones that do [14], the method of Castillo and Jacobs [7] has already been shown to produce significantly better performance.

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.

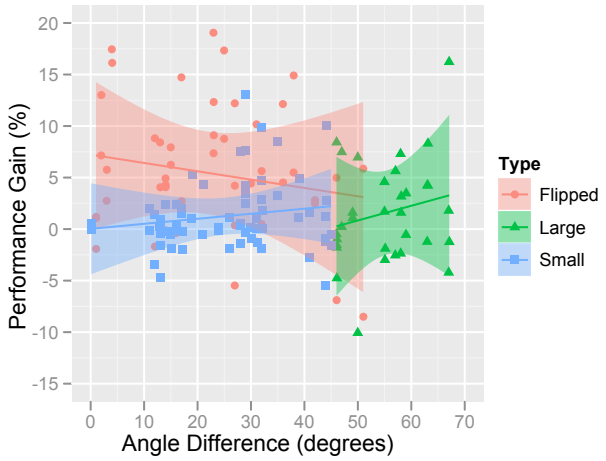

Figure 3. Gain in performance of Slant SMD compared to SMD (in the 68 face test case), as the angle difference changes. Flipped refers to cases where the azimuthal angles of the poses being compared have different signs, unflipped is when the azimuthal angles have the same sign. All bands show a 90% confidence interval.

Table 1. A comparison of recognition accuracy averaged across pose of our slant-compensated stereo matching distance with other methods.

34 Faces

Method	Accuracy
Eigenfaces [14]	16.6%
FaceIt [14]	24.3%
Eigen light-fields (Multi-point norm.) [14]	66.3%
SMD (Castillo and Jacobs [7])	86.8%
Slant SMD	**90.1%**

68 Faces

Method	Accuracy
LiST (Romdhani et al. [21])	74.3%
SMD (Castillo and Jacobs [7])	82.4%
Slant SMD	**85.3%**

5.1. Pose Variation Experiments

A summary of our pose variation experiments is presented in Table 1. 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 3 along with confidence intervals on the prediction.

Table 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.

5.1.1 Statistical Significance

To determine the significance of these results we used McNemar's test [19]. 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 2.

5.2. Pose+Illumination Variation Experiments

Table 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 [21] 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 [21] as one that is favorable for their method; [7] 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 2. Pose variation table for 68 faces comparing the use of the stereo matching method of [7] with our slant-compensated method. Cell format: ⟨accuracy Slant SMD⟩ / ⟨accuracy for SMD [7]⟩. 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 [21] and [15].

azimuth	-66	-47	-46	-32	-17	0	0	0	16	31	44	44	62	
altitude	3	13	2	2	2	15	2	1.9	2	2	2	13	3	
prb. pose	c34	c31	c14	c11	c29	c09	c27	c07	c05	c37	c25	c02	c22	avg
gall. pose														
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

Table 3. Pose + illumination variation for frontal, side and profile probe table using the stereo matching method of Castillo and Jacobs [7], the method of Romdhani et al. [21] 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 2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	avg
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	100	96
	SMD	85	92	95	98	100	98	100	100	98	100	100	100	100	100	100	98	98	100	100	100	100	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

5.3. 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 [7] when the poses are very similar (small variation in pose). The method also works works well in small pose variation cases.

6. 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.

Acknowledgements: This material is based upon work supported by the National Science Foundation under Grant No. 0915977 and US Office of Naval Research Under MURI Grant N00014-08-10638.

References

[1] A. B. Ashraf, S. Lucey, and T. Chen. Learning patch correspondences for improved viewpoint invariant face recognition. In *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2008. 538

[2] R. Basri and D. Jacobs. Lambertian reflectance and

linear subspaces. *IEEE Trans. Pattern Anal. Mach. Intell.*, 25(2):218–233, 2003. 538

[3] P. N. Belhumeur. A Bayesian approach to binocular steropsis. *International Journal of Computer Vision*, 19(3):237–260, 1996. 539, 540

[4] D. Beymer and T. Poggio. Face recognition from one example view. Technical Report AIM-1536, , 1995. 537

[5] S. Birchfield and C. Tomasi. A pixel dissimilarity measure that is insensitive to image sampling. *IEEE Trans. Pattern Anal. Mach. Intell.*, 20(4):401–406, 1998. 538

[6] V. Blanz and T. Vetter. Face recognition based on fitting a 3-D morphable model. *IEEE Trans. Pattern Anal. Mach. Intell.*, 25(9):1063–1074, 2003. 538

[7] C. D. Castillo and D. W. Jacobs. Using stereo matching with general epipolar geometry for 2-D face recognition across pose. *IEEE Trans. Pattern Anal. Mach. Intell.*, 31(12):2298–2304, 2009. 537, 538, 539, 541, 542, 543

[8] X. Chai, S. Shan, X. Chen, and W. Gao. Locally linear regression for pose-invariant face recognition. *IEEE Transactions on Image Processing*, 16(7):1716–1725, 2007. 538, 541

[9] 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. 539

[10] A. Criminisi, A. Blake, C. Rother, J. Shotton, and P. 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. 538, 539, 540

[11] F. Devernay and O. Faugeras. Computing differential properties of 3-D shapes from stereoscopic images without 3-D models. In *CVPR*, 1994. 538

[12] A. Georghiades, P. Belhumeur, and D. 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. 538

[13] R. Gopalan and D. Jacobs. Comparing and combining lighting insensitive approaches for face recognition. *Computer Vision and Image Understanding*, 114(1):135 – 145, 2010. 538

[14] R. Gross, S. Baker, I. Matthews, and T. Kanade. Face recognition across pose and illumination. In S. Z. Li and A. K. Jain, editors, *Handbook of Face Recognition*. Springer-Verlag, June 2004. 541, 542

[15] R. Gross, J. Shi, and J. Cohn. Quo vadis face recognition? In *Third Workshop on Empirical Evaluation Methods in Computer Vision*, December 2001. 543

[16] G. Li and S. W. Zucker. Stereo for slanted surfaces: First order disparities and normal consistency. In A. Rangarajan, B. C. Vemuri, and A. L. Yuille, editors, *EMMCVPR*, volume 3757 of *Lecture Notes in Computer Science*, pages 617–632. Springer, 2005. 538, 539

[17] S. Lucey and T. Chen. A viewpoint invariant, sparsely registered, patch based, face verifier. *International Journal of Computer Vision (IJCV)*, December 2007. 538

[18] J. Matas, O. Chum, M. Urban, and T. Pajdla. Robust wide baseline stereo from maximally stable extremal regions. In *BMVC*, 2002. 537, 538

[19] Q. McNemar. Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika*, 12:153–157, 1947. 542

[20] A. S. Ogale and Y. Aloimonos. Stereo correspondence with slanted surfaces: Critical implications of horizontal slant. In *CVPR (1)*, pages 568–573, 2004. 538

[21] S. Romdhani, V. Blanz, and T. Vetter. Face identification by fitting a 3-D morphable model using linear shape and texture error functions. In *ECCV'02*, volume 4, pages 3–19, Copenhagen, Denmark, 2002. 538, 541, 542, 543

[22] T. Sim, S. Baker, and M. Bsat. The cmu pose, illumination, and expression database. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(12):1615 – 1618, December 2003. 541

[23] L. Wiskott, J.-M. Fellous, N. Krüger, and C. von der Malsburg. Face recognition by elastic bunch graph matching. In *CAIP'97, Kiel*, number 1296, pages 456–463, Heidelberg, 1997. Springer-Verlag. 537

[24] X. Zhang and Y. Gao. Face recognition across pose: A review. *Pattern Recognition*, 42(11):2876–2896, 2009. 537

[25] W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld. Face recognition: A literature survey. *ACM Comput. Surv.*, 35(4):399–458, December 2003. 537

[26] Z. Zhou, A. Ganesh, J. Wright, S.-F. Tsai, and Y. Ma. Nearest-subspace patch matching for face recognition under varying pose and illumination. In *FG*, pages 1–8. IEEE, 2008. 541