Speeding up 2D-Warping for Pose-Invariant Face Recognition

Harald Hanselmann and Hermann Ney
Human Language Technology and Pattern Recognition Group, RWTH Aachen University, Germany
surname@cs.rwth-aachen.de

***Abstract*— Recently, state-of-the-art recognition accuracies for pose-invariant face recognition have been achieved by using 2D-Warping methods in a nearest-neighbor framework. However, the main drawback of these methods is the high computational complexity. In this paper we address this issue. We use a simple and fast method to get a rough estimate of a 2D-Warping. This estimate can then be used to apply an image dependent warprange on the 2D-Warping algorithm, limit the possible poses or preselect the most likely classes. By this method we are able significantly reduce the runtime of a recently proposed 2D-Warping algorithm without sacrificing recognition accuracy.**

I. Introduction

Next to illumination, occlusion and facial expression, pose-variation is one of the main challenges in face recognition. The approaches to pose-invariant face recognition proposed in the literature range from subspace methods [16], [5], [23] over methods based on frontal face reconstruction and 3D-Models [3], [22], [28], [17] to algorithms using 2D-Warping [1], [21], [2], [11], [9]. For the latter, warpings (image matchings) between gallery (train) and test images are computed to generate a warping score that can be used in a nearest neighbor classifier [14]. These methods do not require a training step and can achieve state-of-the-art results with only one gallery image per person. In this context the work in [9] recently reported good results with a fully automatic recognition setup on the CMU-MultiPIE database, one of the most challenging benchmark databases for pose-invariant face recognition. Due to the high computational cost of generating the 2D-Warpings, the runtime is the main drawback of the method in [9]. One possible solution to this problem is parallelization. On one hand the warping algorithm can be implemented using parallelization frameworks such as NVIDIA CUDA [20], on the other hand the computation of the warpings can be distributed across many cores, because they are independent from each other. Another possibility is to reduce the runtime of the 2D-Warping method by adjusting the warping algorithm and classification process. In this paper we compute a fast estimate of the warping and use it to impose restrictions on the computationally expensive warping algorithm originally proposed in [9]. By this method, the latter can be forced not to compute warpings that are unlikely to lead to a successful classification and thus reduce the runtime. Additionally, the computed estimate can be used for pre-selection strategies.

The paper is structured as follows. In Section II we describe the face recognition pipeline used in [9] followed by a review of 2D-Warping in Section III. In Sections IV,V and VI we describe our proposed method. It is evaluated on the CMU-MultiPIE database in Section VIII and we finish with a conclusion and outlook in Section IX.

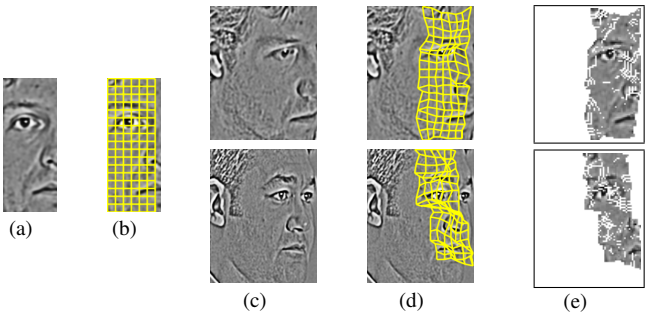


Fig. 1. Examples of 2D warping. Gallery image (a) is mapped to test images (c). In (b) and (d) the deformation grids are shown while (e) shows the deformed gallery image.

II. Pose-Invariant Face Recognition Pipeline

The authors in [9] use a face recognition pipeline that employs fully automatic classification given only a single gallery image per class. The test images are processed by a face detector [12] resulting in a rough bounding-box of the face. The gallery images are cropped and normalized manually. Both, test and gallery images are resized to a resolution of 204×256 pixels and normalized against illumination changes [25]. To classify a given test image, the nearest neighbor rule with 2D-Warping scores as similarity measure is used. This means each gallery image has to be mapped to the test image to get a score for the corresponding class. To avoid occlusions caused by pose variations, the left and the right half of each gallery image is mapped independently as proposed in [1]. The one with the best score is selected for nearest neighbor classification. This is done for all test images, since at test time it is not known whether a pose-induced occlusion is present or not. As a result, the gallery images used for matching are half the size of the test images. An example of the matching between a halved gallery image and different test images is given in Fig. 1. It shows the left half of the gallery image and the test images after preprocessing and face detection. As a result of the preprocessing the images are roughly aligned with respect to the vertical axis but not the horizontal axis. Fig. 1 further shows the result of mapping the gallery image to the test images. It can be observed that the target region (everything covered by the deformed grid in the test image) is a patch of roughly the same size as the gallery image. However, the 2D-Warping algorithm searches the whole image for the optimal solution. This motivates to find a quick approximation of

the target patch in advance and restrict the 2D-Warping algorithm accordingly.

III. 2D-Warping

The goal in 2D-Warping (2D Image-matching) is to find the optimal mapping between two images regarding a specific cost function. In [21] the problem is defined as follows. Given a source image X with dimensions $I \times J$ and a target image R with dimensions $U \times V$ (in this paper we assume $U >= I$ and $V >= J$; the gallery image is the source image while the test image is the target image) one is interested in a warping w that assigns a pixel $w_{ij} = (u, v) \in R$ to each pixel $(i, j) \in X$. This mapping is computed by optimizing the energy function

$$E(X, R, \{w_{ij}\}) = \sum_{i,j} \big[d(x_{ij}, r_{w_{ij}}) + T(w_{i-1,j}, w_{ij}) + T(w_{i,j-1}, w_{ij})\big]. \quad (1)$$

In this equation function $d(\cdot)$ denotes a distance function between two feature vectors $x_{i,j}$ and $r_{w_{ij}}$ representing pixel (i, j) and w_{ij}. The distance function can also be extended to include a small local context, e.g. instead of computing the distance between the feature vectors of two pixels directly the distances within a squared window around the pixels are accumulated and normalized by the window size. A cutoff-threshold τ can be used on the local distances to be more robust against outliers. The relative penalty function $T(\cdot)$ is the smoothness term. Different displacement vectors of neighboring pixels are penalized by using the Euclidean distance on the warping positions weighted by a factor α. Additionally monotonicity and continuity constraints are integrated [27].

Due to the 2D-dependencies, optimizing Formula (1) is NP-complete [15]. However, there are several algorithms available to find an approximative solution [14], [1], [21], [9]. Two of them are used in this work, namely Zero-Order Warping (ZOW) [14] and Two-Level Dynamic Programming with Lookahead (2LDP-LA) [9].

A. Zero-Order Warping

Zero-Order Warping (also referred to as Image Distortion Model) [14], [21] discards all 2D-dependencies and optimizes each pixel independently. As a result, no relative penalty function can be defined. To apply at least some kind of restriction on the warping, an absolute penalty $T_\Delta(\cdot)$ is used:

$$E(X, R, \{w_{ij}\}) = \sum_{i,j} \big[d(x_{ij}, r_{w_{ij}}) + T_\Delta((i, j), w_{ij})\big], \quad (2)$$

where

$$T_\Delta((i, j), w_{ij} = (u, v)) = \begin{cases} \alpha d_{pen}((i, j), w_{ij}) & \text{if } |v - j| \le \Delta \wedge |u - i| \le \Delta \\ \infty & \text{else} \end{cases} \quad (3)$$

$T_\Delta(\cdot)$ is based on the absolute displacement of pixel (i, j) and penalizes deviations by using a distance function $d_{pen}(\cdot)$. Additionally an absolute warprange Δ is included as an upper bound for possible displacements. This approach is only promising if the the two images to be matched are already roughly aligned and the dimensions are normalized.

With a complexity of $\mathcal{O}(IJ(2\Delta + 1))$ ZOW is quite fast, but since neighboring pixels are mapped independently it can not capture image structures. Warprange, absolute penalty and local context can only compensate for that to a limited extend.

B. Two-Level Dynamic Programming

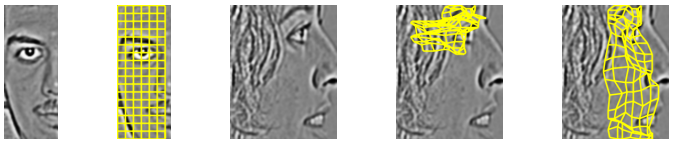
In contrast to ZOW, the Two-Level Dynamic Programming algorithm introduced in [9] does not need to discard the 2D-dependencies between the pixels. Instead, an approximative optimization of Formula (1) is done by sequentially optimizing the columns over a representative row of the 2D-grid, guided by a lookahead (2LDP-LA). This method has been applied to pose-invariant face recognition and achieved state-of-the-art results for a fully automatic recognition setup (see Section II). However, the computation is still complex and time consuming (the authors report 16 seconds for one warping). One reason for this is that the optimization is unrestricted in the sense that each pixel from the source image can be mapped to any position in the target image. This is a nice property in general, but unnecessary in the case of face recognition. For example, a warping where all pixels are mapped to the same target pixel is possible and considered by 2LDP-LA. But if this warping would be optimal, then it is very unlikely that this leads to a successful recognition. Therefore, we do not need to explore this option in the first place. This can be achieved by introducing a warprange as in ZOW [14]. However, as in ZOW, an absolute warprange leads to the problem that the images have to be roughly aligned, which is not the case in general. For this reason we use an image dependent warprange which is described in the next section.

IV. Image Dependent Warprange

Using a warprange to reduce the search space for possible warping mappings is often used, e.g. for ZOW [14], Tree-Serial Dynamic Programming [19] or the maximal disparity in stereo vision [24]. As mentioned before, applying an absolute warprange requires the images to be roughly aligned. If pixel (i, j) is forced to be mapped to the region $(i \pm \Delta, j \pm \Delta)$ for a small Δ, then a larger translation makes reaching the optimal target patch impossible. For this reason, we use a warprange that depends on the source and target image at hand. This is done by altering Formula (1) to include an additional penalty function defining the warprange δ

$$E(X, R, \{w_{ij}\}) = \sum_{i,j} \big[d(x_{i,j}, r_{w_{ij}}) + T(w_{i-1,j}, w_{ij}) + T(w_{i,j-1}, w_{ij}) + T_\delta((i, j), w_{ij})\big], \quad (4)$$

where



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

Fig. 2. Example of image matching with 2LDP-LA (d) and 2LDP-LA-W (e). Gallery image (a) is matched to test image (c).

$$T_\delta((i,j), w_{ij} = (u,v)) = \begin{cases} 0 & \text{if } |v-j| \leq \delta_v + o_v^{X,R} \\ & \wedge |u-i| \leq \delta_h + o_h^{X,R} \\ \infty & \text{else} \end{cases} \quad (5)$$

Here we divide δ into a horizontal δ_h and vertical δ_v component to have separate warpranges for both image dimensions. The variables $o_h^{X,R}$ and $o_v^{X,R}$ define a relative offset for the warprange. They are dependent on the two images to be matched (X and R) and are computed in advance. This is done by using a much faster warping method to get a rough estimate of the right target region for image X in image R. For this purpose a sliding window variant of ZOW (described in the next section) or facial landmarks can be used.

In practice, warpings that lead to $T_\delta(\cdot) = \infty$ are excluded from the optimization. Since the complexity of 2LDP-LA heavily depends on the search space UV for each pixel (for details see [9]), reducing this to $(2\delta_h + 1)(2\delta_v + 1)$ significantly improves the complexity for low values of δ_h and δ_v. Note that this image dependent warprange still assumes that the dimensions of the source image and the correct patch in the target image are similar, since we are doing no scale normalization. However, small differences as they are present in the face recognition pipeline considered here (the face detection step also leads to roughly normalized scales) can be compensated by adjusting the warpranges. Since we are using 2LDP-LA in our approach, we call the extension 2LDP-LA-W. It should be noted that the same speedup methods can also be used on other 2D-Warping algorithms that optimize Formula (1) in a similar setup.

Additional to the relative penalty and the monotonicity and continuity constraints, the warprange introduces another incentive for smooth warpings. E.g. while a warprange of 0 would disallow all deformations, a moderate warprange only disallows extreme warpings such as mapping all source pixels to the same target pixel. This is illustrated in Fig. 2. It shows an example of a case where 2LDP-LA fails to compute a smooth warping, but 2LDP-LA-W on the other hand generates a better result due to the the forced smoothness by the warprange.

V. Target Patch Approximation

To approximate the target patch and determine the relative offsets for 2LDP-LA-W we consider two options, which are introduced in the following.

A. Sliding Window ZOW

To apply ZOW when the images are not aligned is problematic, because on one hand a large warprange must be chosen to compensate for large translations but on the other hand a large warprange leads to non-smooth warpings which lead to bad classification results. To overcome this problem we apply ZOW in a sliding window (SW-ZOW), similar to template matching (e.g. [10]). This is done in a pseudo-2D fashion using a two-pass procedure, since a complete search would be too computationally expensive. In the first pass, a horizontal sliding window is used while the vertical offset is fixed to zero. The horizontal offset that leads to the best match is then fixed and a vertical sliding window is used to refine the result. We do the horizontal pass first based on the assumption that the images are already roughly aligned in the vertical dimension (Section II). Note that the sliding window does not define a strict bounding box and pixels can be warped outside the window as far as the warprange Δ allows.

The complexity of SW-ZOW is as follows. Assuming $U >= I$ and $V >= J$ the minimal horizontal offset for $o_h^{X,R}$ is given by $-\Delta$ and the maximum is $U - I + \Delta$. Therefore the offset in the horizontal dimension can take a value in the range of $[-\Delta, ..., U - I + \Delta]$. For each of these offsets a ZOW-model has to be optimized. As mentioned in Section III-A the complexity for ZOW is $\mathcal{O}(IJ(2\Delta + 1))$, not considering boundary effects. As a result the complexity for the horizontal pass of the sliding window ZOW is $\mathcal{O}((U - I + 2\Delta + 1)IJ(2\Delta + 1))$. The same considerations apply for the vertical pass which means in total ZOW has to be applied $(U-I+V-J+4\Delta+2)$ times. This leads to a final complexity of $\mathcal{O}((U-I+V-J+4\Delta+2)IJ(2\Delta+1))$. This complexity has to be added to 2LDP-LA-W as overhead, but the results in Section VIII show that overall the runtime is improved.

Applying ZOW in this manner can be interpreted as trying different offsets for 2LDP-LA but instead of computing expensive warpings with 2D-dependencies the dependencies are discarded to get a fast estimate of the quality of the offset. In other words, while 2LDP-LA searches the image while maintaining expensive 2D-dependencies, SW-ZOW searches the image without them.

B. Landmarks

The SW-ZOW is useful to find the most likely matching region when nothing is known about the test image. However, if facial landmarks are available (e.g. manual annotations or by applying a landmark detector such as [30]), they can be used directly to locate the best region in the test image. E.g. from Fig. 1 it can be observed that the landmarks eye and mouth are mapped onto each other. If the landmarks are available at test time, then the distances of the landmark positions between gallery and test image can be used to determine the offset for 2LDP-LA-W by taking the average for all landmarks. A fast estimate of the optimal 2D-Warping can then be obtained by applying ZOW once at the determined offset or applying SW-ZOW in a small sliding

window around the offset to weaken the effect of inaccurate landmarks.

A common preprocessing step for face recognition is to apply a pre-alignment procedure that uses landmarks and pose information to normalize rotation and scale of the faces (e.g. the work in [5]). Such techniques are also helpful for 2LDP-LA-W, since pre-aligned images allow the use of tighter warpranges.

VI. Pre-selection

In the previous sections the goal was to reduce the complexity of a single 2LDP-LA warping. However, for the classification using 2LDP-LA in a nearest-neighbor framework, several such warpings have to be computed. Given C classes and the face recognition pipeline from Section II, $2C$ different 2LDP-LA warpings are needed to classify an image (left and right half of each gallery image are matched to the test image). This number can be significantly reduced by pre-selection techniques similar to pre-filtering [26]. To be more precise, we use two pre-selection strategies.

First, we can decide early on whether the left or the right half of the gallery image is better suited for classification (LR pre-selection). In case the pose of the test image is known we also know which half of the face is not occluded (based on the sign of the pose angle) and warp the respective half of the gallery image. In case the pose angle is 0 we warp the left half. In case we do not have any pose information we compute the SW-ZOW score for both halves and take the one with the lower score.

Second, we can use the SW-ZOW energies to eliminate unlikely classes by ranking the gallery images according to their SW-ZOW scores and keeping only the best k for classification with 2LDP-LA (class pre-selection), similar to [8].

VII. Technical details

We extended the implementation used in [9] and [6] and follow the proposition in [29] by using integral images to compute the local contexts of the distance function. Additionally, we used the NVIDIA CUDA framework [20] to implement a parallel version of the 2LDP-LA warping algorithm[1]. The runtime measurements presented in Section VIII were performed using a notebook with a 2.80 GHz Intel Core i7-3840QM CPU and a NVIDIA Quadro K2000M GPU averaging over ten representative warpings, while the recognition results were obtained using a cluster with over 100 AMD CPUs. For the latter we observed significantly higher runtimes for single warpings than on the notebook, but comparable speedup factors.

VIII. Experimental Evaluation

We evaluate the proposed method on the CMU-MultiPIE [7] database, which is a popular benchmark for pose-invariant face recognition. The database has 337 classes and the images were recorded over four different recording sessions. Amongst others, the database offers variations in pose from $-90°$ to $+90°$ yaw (13 different poses in total). Additionally there are two poses (08_1 and 19_1) where next to $45°$ yaw also a slight variation in pitch is present to simulate the typical angle of a surveillance camera. In the original database the images for the two surveillance poses are delivered upside down, therefore we rotate them by $180°$ prior to all our experimental setups.

A common approach for pose-invariant face recognition is to use the first 100 subjects from the first session for model training and the remaining 149 subjects from the same session for testing. The illumination is constant in all images (flash 07) and pose 05_1 is used as gallery image (one image per subject), while the other 14 poses are used for testing (overall 2086 test images). Since our classification approach does not involve a training step, we use the first 100 subjects (specifically pose 11_0 ($-90°$)) to tune our basic warping parameters: The warping penalty α, the distance threshold τ and the size of the local context. As in [9] we use PCA reduced U-SIFT features [18], [13], [4].

We use two basic setups in our experiments.

1) Face detection setup: First we use the same pipeline as in [9], which has already been described in Section II. In this case no manual annotations are used for the test image and the recognition is done fully automatic. Only the gallery images are normalized using the annotations and alignment procedure from [5]. This setup is well suited to evaluate the performance of 2LDP-LA-W in combination with SW-ZOW, but since no landmarks are available we cannot test 2LDP-LA-W in combination with facial landmarks.

2) Landmark setup: To evaluate 2LDP-LA-W with landmarks to localize the target region we use a second setup where facial landmarks are given for all images. Additionally we require the yaw and pitch angles to be known in order to perform the alignment proposed in [5] also for the test images. For this purpose we use the manual annotations from [5] to see the full potential of our method. However, as a result, the experiments for this setup are not fully automatic. Thus we repeat the experiments using a publicly available landmark detector and pose estimator [30]. For simplicity we use one of the pre-trained models supplied by the authors. Running a landmark detector and pose estimator causes additional computation time but has to be done only once for each test image and not for each test and gallery image pair.

It should be noted that we only compare the runtime of the matching algorithms in this section. The additional steps (face detection / landmark detection, preprocessing, feature extraction etc.) are not considered.

A. Face detection setup

Table I shows the results for 2LDP-LA-W with SW-ZOW and different values for the warprange of SW-ZOW Δ. Additionally we give the runtime and speedup factor compared to the baseline 2LDP-LA for one image matching. The runtimes for 2LDP-LA-W are given *including* the runtime for SW-ZOW. The latter can also be used directly for classification

[1]Implementations available at
`http://www.hltpr.rwth-aachen.de/w2d`

TABLE I

RESULTS ON THE CMU-MULTIPIE DATABASE FOR DIFFERENT VALUES OF Δ WITH $\delta_h = 20$ AND $\delta_v = 15$ FIXED.

Method	Δ	time per warping[s]	speedup factor	Acc[%]
2LDP-LA	-	8.88	-	86.7
SW-ZOW	8	1.44	6.2	78.0
	7	1.21	7.3	79.0
	6	0.96	9.3	77.9
	5	0.77	11.5	75.9
2LDP-LA-W	8	3.18	2.8	86.8
	7	2.95	3.0	86.6
	6	2.72	3.3	86.6
	5	2.53	3.5	86.4

TABLE II

RESULTS ON THE CMU-MULTIPIE DATABASE FOR DIFFERENT VALUES OF δ_h AND δ_v WITH $\Delta = 6$ FIXED.

Method	δ_h	δ_v	time per warping[s]	speedup factor	Acc[%]
2LDP-LA	∞	∞	8.88	-	86.7
SW-ZOW	-	-	0.96	9.3	77.9
2LDP-LA-W	20	20	3.21	2.8	86.4
	20	15	2.72	3.3	86.6
	15	20	2.68	3.3	86.4
	15	15	2.31	3.8	86.7
	15	10	1.93	4.6	85.9
	10	15	1.88	4.7	86.2
	10	10	1.61	5.5	86.2

TABLE III

RESULTS WITH LR PRE-SELECTION BASED ON SW-ZOW SCORES ON THE CMU-MULTIPIE DATABASE.

Method	speedup factor	Acc[%]
2LDP-LA-W	3.8	86.7
+ LR pre-selection	5.4	86.4

and the accuracy of SW-ZOW is remarkable considering it is a much simpler and faster warping method. However, adding 2LDP-LA-W gives a classification performance boost of almost 10% absolute. For the different values of Δ (the warpranges δ_h and δ_v are fixed to 20 and 15) the recognition accuracy of 2LDP-LA-W is around 86% and very close to the baseline accuracy of 2LDP-LA.

As a compromise between accuracy and speed we select $\Delta = 6$ and do further experiments with different values for δ_h and δ_v. The results are shown in Table II. Also for these experiments the recognition accuracy is around 86%. For $\delta_h = 15$ and $\delta_v = 15$ we even achieve the same accuracy as the baseline, although almost four times faster. Note that applying only SW-ZOW with $\Delta = 6$ leads to a speedup factor of 9.3. This defines an upper bound for the speedup that can be achieved adding 2LDP-LA-W, since SW-ZOW still has to be applied for all gallery images.

Table III gives results for the pre-selection of the left or right half of the gallery image based on SW-ZOW scores (see Section VI). While the recognition accuracy drops by 0.3% absolute, the speedup factor compared to 2LDP-LA is now at 5.4 (based on the runtimes in Table II).

TABLE IV

RESULTS ON THE CMU-MULTIPIE DATABASE WITH LANDMARKS.

Method	Target Approximation	time per warping[s]	speedup factor	Acc[%]
2LDP-LA	-	8.88	-	90.5
2LDP-LA-W	SW-ZOW	1.40	6.3	90.9
2LDP-LA-W	Manual LM	0.44	20.2	91.5
2LDP-LA-W	Autom. LM + SW-ZOW	0.90	9.9	88.8

B. Landmark setup

In Table IV the results with landmarks are given. We only use the landmarks of the eyes to find the optimal target patch since preliminary experiments have shown this to be optimal. Note that for the results in Table IV we already applied LR pre-selection based on the sign of the pose angles (see Section VI) for all methods including the baseline 2LDP-LA. It is no surprise that with manual annotations and pre-alignment the baseline accuracy is improved to 90.5%. For 2LDP-LA-W we keep $\Delta = 6$, but use lower values for δ_h and δ_v, since the images are pre-aligned. We set the parameters to 10 and 6, respectively. Using 2LDP-LA-W with full SW-ZOW as in the previous section already leads to a small improvement of the recognition accuracy compared to 2LDP-LA while being six times faster. Completely skipping SW-ZOW and using only the landmark information to approximate the target patch for 2LDP-LA-W leads to another increase in accuracy and this setup is even 20 times faster. These two results indicate that on one hand, restricting 2LDP-LA by a warprange leads to better classification results and on the other hand SW-ZOW does not always find the optimal target patch.

For the automatic landmark detector we use SW-ZOW in a window of 13 pixels around the detected landmark explaining the increased runtime compared to manual landmarks. With this setting, 2LDP-LA-W reaches an accuracy of 88.8%, which is not only better than the best result in an automatic setup so far (see Table II), but also very close to the results achieved with manual annotations. In fact, in Table V we can see that there is only a large difference for the surveillance images, which is due to a high number of errors by the landmark detector (the pre-trained models [30] were not trained on images with tilt).

C. Class pre-selection

In Fig. 3 the effect of pre-selecting the best k classes using SW-ZOW scores on the recognition accuracy is illustrated. Results are given for both experimental setups and all results are after the respective LR pre-selection. For the face detection setup with $k = 149$ the accuracy corresponds to the result in Table III. When decreasing k the accuracy is constant at around 86% until with $k = 36$ the accuracy drops by 1% absolute compared to 2LDP-LA. The overall speedup factor at this point is 7.8 (SW-ZOW is still done

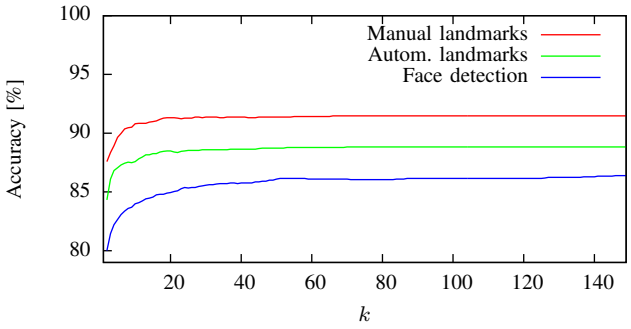


Fig. 3. Recognition accuracy for 2LDP-LA-W with class pre-selection by using the SW-ZOW scores.

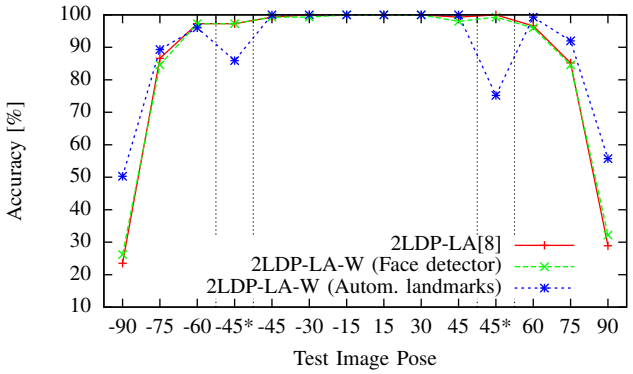


Fig. 4. Recognition accuracies for each pose for fully automatic methods. The surveillance poses are marked with 45^* and -45^*.

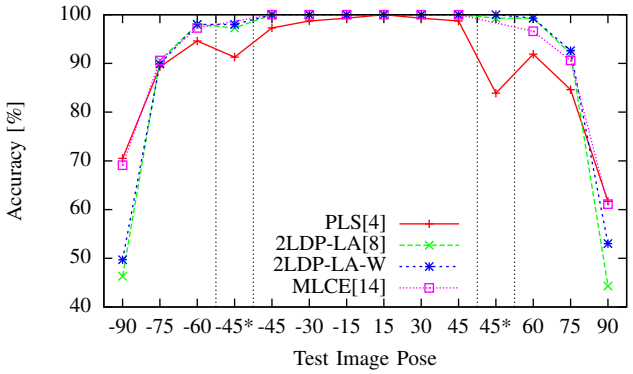


Fig. 5. Recognition accuracies for each pose for methods using manual annotations at test time. The surveillance poses are marked with 45^* and -45^*.

for all classes, but 2LDP-LA-W only for 36 classes). As a result, putting everything together we can achieve a speedup factor of 3.8 without losing any classification performance and a speedup factor of 7.8 if we allow the performance to degrade by 1% absolute.

In case of the manual landmarks the runtime without SW-ZOW is 0.44s (see Table IV). In order to get the scores for the class pre-selection we run ZOW once at the offset defined by the landmarks, which slightly increases the runtime to 0.55s. Based on these numbers and the observation that the recognition accuracy is constant at 91.5% for $k >= 66$, we can achieve a speedup factor of 29 for this experiment without losing accuracy. At $k < 9$ the accuracy drops by more than 1% absolute. However, this is still as good as the baseline and the speedup factor for $k = 9$ equals 65.

The results for automatic landmarks are similar. Here the accuracy stays at 88.8% for $k >= 70$ and above 87.8% for $k >= 12$. However, the respective speedup factors are only 13 and 18, since SW-ZOW in a window around the detected landmark takes longer than ZOW at a fixed position.

D. Parallel implementation

For the previous experiments the runtimes were computed using one CPU core, but the used warping algorithm is also parallelizable. To get an idea of how much this can improve the runtime we compare the CPU implementation with a parallel GPU implementation using the NVIDIA CUDA framework. Where a full 2LDP-LA warping took 8.88 seconds on the CPU, we measured 1.37 seconds on the GPU. Similarly the runtime for 2LDP-LA-W in combination SW-ZOW ($\Delta = 6$, $\delta_h = 15$ and $\delta_v = 15$) was improved from 2.31 seconds to 0.6 seconds. Note that due to numerical differences between platforms and implementations (e.g. double vs. single precision floating point numbers) the computed energies differ and can therefore also alter the classification results.

E. Comparison to the State-of-the-Art

In Table V (and more detailed in Fig. 4 and 5) we compare the recognition performance of 2LDP-LA-W with some state-of-the-art methods that use a similar setup. We differentiate between methods using manual landmarks and methods using a fully automatic testing procedure and group the images into three sets, near-frontal and near-profile with six images each, and the two surveillance images. While many methods reach perfect or close to perfect performance on the near-frontal images, the near-profile and surveillance images are more difficult. To the best of our knowledge the maximum likelihood correspondence estimation (MLCE) [17] reports the best accuracy for the near-profile images, but the authors give no results for the surveillance images. MLCE as well as the partial least squares method (PLS) from [5] achieve better results on the near-profile images than our method, but it should be noted that MLCE and PLS both train pose specific models. This is in contrast to our method since we use only one parameter set for all poses in order to be more robust against inaccurate landmark and pose information. However, it is also possible to optimize pose-specific parameter sets to improve the accuracy.

To the best of our knowledge, 2LDP-LA-W in combination with automatic landmarks achieves with 88.8% accuracy the best performance for methods with a fully automatic recognition setup. This is also remarkably close to the approaches using manual annotations.

TABLE V

COMPARISON WITH STATE-OF-THE-ART RECOGNITION ACCURACIES.

Method	Manual test image annotations	Near-frontal[%]	Near-profile[%]	Surveillance[%]	Total without[%] surveillance	Total[%]
2LDP-LA [9]	no	99.78	69.69	98.70	84.73	86.72
2LDP-LA-W (Face detector)	no	99.44	70.13	98.30	84.79	86.72
2LDP-LA-W (Landmark detector)	no	100.00	80.43	80.54	90.21	88.83
2LDP-LA [9]	yes	100.00	78.41	98.32	89.21	90.51
2LDP-LA-W	yes	100.00	80.43	98.99	90.21	91.47
PLS [5]*	yes	98.88	82.10	87.60	90.50	90.07
MLCE [17]	yes	100.00	84.22	-	92.11	-

* Different illumination (flash 00).

IX. CONCLUSION

In this paper we have integrated an image dependent warprange to the 2D-Warping algorithm 2LDP-LA. This warprange is found by either applying ZOW in a sliding window, facial landmarks or a combination of both. The experiments on a pose subset of the CMU-MultiPIE have shown that by this approach, the runtime can be signficantly improved by factors ranging from 3.8 to 29 for various setups without sacrificing accuracy (even higher when we tolerate slightly lower accuracies). On the contrary, we are able to improve the baseline accuracy for a setup with manual annotations as well as a fully automatic setup including a landmark detection and pose estimation step. To the best of our knowledge, the result of 88.8% accuracy is the best result achieved so far in a fully automatic setup.

While SW-ZOW is much faster than 2LDP-LA, there is still room for improvement. For example the step-size for the sliding window could be adjusted or different scales for the window could be used. The latter would make the method more interesting for other use cases than the face recognition pipeline used in this work.

REFERENCES

[1] S. Arashloo, J. Kittler, and W. Christmas. Pose-invariant face recognition by matching on multi-resolution mrfs linked by supercoupling transform. *Computer Vision and Image Understanding*, 115(7):1073–1083, 2011.

[2] S. R. Arashloo and J. Kittler. Fast pose invariant face recognition using super coupled multiresolution markov random fields on a gpu. *Pattern Recognition Letters*, 48:49–59, 2014.

[3] A. Asthana, T. Marks, M. Jones, K. Tieu, and M. Rohith. Fully automatic pose-invariant face recognition via 3d pose normalization. In *IEEE ICCV*, pages 937–944, 2011.

[4] P. Dreuw, P. Steingrube, H. Hanselmann, and H. Ney. Surf-face: Face recognition under viewpoint consistency constraints. In *BMVC*, 2009.

[5] M. Fischer, H. K. Ekenel, and R. Stiefelhagen. Analysis of partial least squares for pose-invariant face recognition. In *IEEE BTAS*, pages 331–338, 2012.

[6] T. Gass, L. Pishchulin, P. Dreuw, and H. Ney. Warp that smile on your face: optimal and smooth deformations for face recognition. In *FG*, pages 456–463, 2011.

[7] R. Gross, I. Matthews, J. Cohn, T. Kanade, and S. Baker. Multi-pie. *Image and Vision Computing*, 28(5):807–813, 2010.

[8] H. Hanselmann and H. Ney. Fine-grained visual categorization with 2d-warping. In *ICPR*, pages 608–613, 2014.

[9] H. Hanselmann, H. Ney, and P. Dreuw. Pose-invariant face recognition with a two-level dynamic programming algorithm. In *Pattern Recognition and Image Analysis*, pages 11–20. Springer, 2013.

[10] S. Hinterstoisser, C. Cagniart, S. Ilic, P. Sturm, N. Navab, P. Fua, and V. Lepetit. Gradient response maps for real-time detection of textureless objects. *IEEE T-PAMI*, 34(5):876–888, 2012.

[11] H. T. Ho and R. Chellappa. Pose-invariant face recognition using markov random fields. *IEEE Transactions on Image Processing*, 22(4):1573–1584, 2013.

[12] Z. Kalal, J. Matas, and K. Mikolajczyk. Weighted sampling for large-scale boosting. *BMVC*, 2008.

[13] Y. Ke and R. Sukthankar. Pca-sift: A more distinctive representation for local image descriptors. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 2, pages II–506, 2004.

[14] D. Keysers, T. Deselaers, C. Gollan, and H. Ney. Deformation models for image recognition. *IEEE T-PAMI*, pages 1422–1435, 2007.

[15] D. Keysers and W. Unger. Elastic image matching is NP-complete. *Pattern Recognition Letters*, 24(1-3):445–453, 2003.

[16] A. Li, S. Shan, and W. Gao. Coupled bias–variance tradeoff for cross-pose face recognition. *IEEE Transactions on Image Processing*, 21(1):305–315, 2012.

[17] S. Li, X. Liu, X. Chai, H. Zhang, S. Lao, and S. Shan. Maximal likelihood correspondence estimation for face recognition across pose. *IEEE Transactions on Image Processing*, 23(10):4587–4600, 2014.

[18] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004.

[19] V. Mottl, A. Kopylov, A. Kostin, A. Yermakov, and J. Kittler. Elastic transformation of the image pixel grid for similarity based face identification. In *ICPR*, pages 549–552, 2002.

[20] J. Nickolls, I. Buck, M. Garland, and K. Skadron. Scalable parallel programming with cuda. *Queue*, 6(2):40–53, 2008.

[21] L. Pishchulin, T. Gass, P. Dreuw, and H. Ney. Image warping for face recognition: From local optimality towards global optimization. *Pattern Recognition*, 45(9):3131–3140, 2012.

[22] U. Prabhu, J. Heo, and M. Savvides. Unconstrained pose-invariant face recognition using 3d generic elastic models. *IEEE T-PAMI*, 33(10):1952–1961, 2011.

[23] A. Sharma, A. Kumar, H. Daume, and D. W. Jacobs. Generalized multiview analysis: A discriminative latent space. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2160–2167, 2012.

[24] R. Szeliski, R. Zabih, D. Scharstein, O. Veksler, V. Kolmogorov, A. Agarwala, M. Tappen, and C. Rother. A comparative study of energy minimization methods for markov random fields with smoothness-based priors. *IEEE T-PAMI*, 30(6):1068–1080, 2008.

[25] X. Tan and B. Triggs. Enhanced local texture feature sets for face recognition under difficult lighting conditions. *AMFG*, pages 168–182, 2007.

[26] L. Torresani, M. Szummer, and A. Fitzgibbon. Learning query-dependent prefilters for scalable image retrieval. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2615–2622, 2009.

[27] S. Uchida and H. Sakoe. A monotonic and continuous two-dimensional warping based on dynamic programming. In *ICPR*, pages 521–524, 1998.

[28] D. Yi, Z. Lei, and S. Z. Li. Towards pose robust face recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3539–3545, 2013.

[29] K. Zhang, J. Lu, and G. Lafruit. Cross-based local stereo matching using orthogonal integral images. *IEEE Transactions on Circuits and Systems for Video Technology*, 19(7):1073–1079, 2009.

[30] X. Zhu and D. Ramanan. Face detection, pose estimation, and landmark localization in the wild. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2879–2886, 2012.