

Fast Image Alignment in the Fourier Domain

Ahmed Bilal Ashraf
Carnegie Mellon University
bilal@cmu.edu

Simon Lucey
Carnegie Mellon University
slucey@cs.cmu.edu

Tsuhan Chen
Cornell University
tsuhan@ece.cornell.edu

Abstract

In this paper we propose a framework for gradient descent image alignment in the Fourier domain. Specifically, we propose an extension to the classical Lucas & Kanade (LK) algorithm where we represent the source and template image’s intensity pixels in the complex 2D Fourier domain rather than in the 2D spatial domain. We refer to this approach as the Fourier LK (FLK) algorithm. The FLK formulation is especially advantageous, over traditional LK, when it comes to pre-processing the source and template images with a bank of filters (e.g., Gabor filters) as: (i) it can handle substantial illumination variations, (ii) the inefficient pre-processing filter bank step can be subsumed within the FLK algorithm as a sparse diagonal weighting matrix, (iii) unlike traditional LK the computational cost is invariant to the number of filters and as a result far more efficient, (iv) this approach can be extended to the inverse compositional form of the LK algorithm where nearly all steps (including Fourier transform and filter bank pre-processing) can be pre-computed leading to an extremely efficient and robust approach to gradient descent image matching. We demonstrate robust image matching performance on a variety of objects in the presence of substantial illumination differences with exactly the same computational overhead as that of traditional inverse compositional LK during fitting.

1. Introduction

The Lucas & Kanade (LK) algorithm [19] is the method of choice for direct¹ alignment of a source image with a pre-defined template image, although many variations upon the basic idea now exist in computer vision literature [4]. The canonical LK approach, however, has serious problems when the template differs from the source image in terms of

¹It should be noted that there are, broadly speaking, two approaches to image alignment: (i) *feature based* approaches rely on abstracting the image by the geometric location of a set of chosen distinctive features, and (ii) *direct* approaches use all the pixels of interest. This paper is concerned with direct approaches to image matching.

appearance (e.g., changes in illumination, intra-class object variation, etc.) [16, 2].

One common approach for dealing with these types of appearance changes is to apply a weighting matrix to the sum of squared distances (SSD) objective function being optimized within the LK algorithm. An excellent treatment on this topic can be found in Baker et al. [2]. We refer to this extension as the weighted LK algorithm. The basic intuition behind this method is that: (i) not all pixels may be equally important for alignment, and (ii) pixels in general have some correlation with one another. A fundamental problem, however, with the weighted LK algorithm lies in the choice of the weighting matrix. For computational reasons, it is rare to use a full rank weighting matrix as each iteration of the traditional LK algorithm would require the multiplication of a $N \times N$ matrix where N is the number of pixels (typically a large number e.g. $N > 10,000$). As a result only low rank or diagonal matrices are used as weighting matrices in practice [2] severely limiting the expressiveness of the weighting matrix. Good performance can, however, still be obtained under these constraints where one adaptively updates the weighting matrix at each iteration (e.g., robust error functions [8]).

This lack of expressiveness becomes especially problematic, however, if one wants to employ the weighted LK algorithm within the extended inverse compositional framework of Baker and Matthews [3]. The inverse compositional algorithm has been shown to be one of the most reliable and computationally efficient registration methods in the vision community. The efficiency stems from the so-called inverse compositional trick, making constant the Hessian matrix involved in the linear least squares problem to be solved at each iteration. To take advantage of the pre-computation with the weighted LK algorithm, however, the weighting matrix must remain constant over all iterations.

In this paper we make a case for posing the LK algorithm in the 2D Fourier domain. The usefulness of such a framework becomes apparent if one explores the problem of applying the traditional LK algorithm simultaneously across a bank of filter image responses. In the spatial domain the complexity of this algorithm is a direct function of the

number of filter banks being applied, greatly increasing the computational and memory requirements of image matching on raw pixels. Other areas of computer vision have dealt with this computational and memory overhead through approximation methods [24, 18, 6, 17] (e.g., subsampling responses, feature selection, etc.).

In our work we instead argue to reformulate the problem by expressing the template and source image pixel intensities in the 2D Fourier domain. Through this reformulation one can instead solve this problem exactly, without any need for approximations, using an algorithm we refer to as Fourier LK (FLK). The FLK algorithm has a number of useful characteristics compared to the traditional LK algorithm namely: (i) computational complexity is *independent* of the number of filter banks employed, and (ii) the pre-processing step can be subsumed within the weighted FLK objective function. Our key contributions in this paper are as follows,

- Exploiting *Parseval’s relation* [21] (energy contents of images are conserved between spatial and Fourier domains) we demonstrate that the LK alignment problem can equivalently be cast in the Fourier domain (FLK) (Section 5).
- We demonstrate that doing image alignment on a high dimensional bank of filter response images is mathematically equivalent to doing alignment in the low dimensional raw image pixel space, if appropriate weightings are applied in the Fourier domain. This formulation opens up the possibility of applying a much richer class of weighting matrices (full rank, non-diagonal) for the weighted LK algorithm (Section 5).
- We demonstrate the utility of our novel FLK algorithm by showing its robustness to illumination changes in comparison to conventional LK (Section 6).
- Further, we derive the inverse-compositional extension for the FLK algorithm and show that it significantly speeds up performance in comparison to the traditional forwards-additive formulation and has exactly the same computational load as that of inverse-compositional LK algorithm in the spatial domain (Section 7).

2. Related work

One of the earliest attempts to handle illumination variation, within the LK framework, was proposed by Hager and Belhumeur [16]. In their approach the authors model the illumination variation by a set of basis vectors derived from training images of the target, taken under varying illumination conditions. Most notably, Black and Jepson addressed the problem for the case of general appearance variation [8]. Baker et al. [2] presented a unifying framework in which much of this previous work could be subsumed and applied within an inverse compositional framework for

computational efficiency. Discriminative approaches within the LK framework have also been explored by Avidan [1] and others [20]. A fundamental drawback to all these approaches, however, is that they require multiple examples of the object under changing illumination.

Robust error functions were also proposed by Black and Jepson [8] for use within the LK framework and have proven useful in the presence of illumination variation without the need for a priori knowledge of the object under different illuminations. An excellent treatment on a variety of robust error functions for a LK fitting task can be found in [23]. A fundamental drawback of the robust error function approach, however, stems from the need to re-estimate the Hessian matrix at each iteration. More recently, a dual inverse compositional framework was proposed by Bartoli [7] where both the geometry as well as the gain and bias of the source image’s appearance can be solved using an inverse compositional framework. This approach, although computationally efficient, has fundamental limitations with respect to illumination change as it can only handle fluctuations in the object’s gain and bias.

3. Weighted Lucas-Kanade

The LK algorithm [4, 19] attempts to find the parametric warp \mathbf{p} that minimizes the SSD between a template image T and a warped source image I , given by the following error term,

$$E = \| I(\mathbf{p}) - T(\mathbf{0}) \|^2 \quad (1)$$

where $I(\mathbf{p})$ represents the warped input image using the warp specified by the parameters \mathbf{p} , while $T(\mathbf{0})$ represents the unwarped template image. In the original work of Lucas and Kanade [19] the parameter vector \mathbf{p} represented translation, but in principle one may represent any other type of parametric warp such as affine [8, 4] or piece-wise affine [14].

The weighted LK algorithm proposed by Baker and Matthews [2] attempts to find a warp that minimizes the following error,

$$E = \| I(\mathbf{p}) - T(\mathbf{0}) \|_{\mathbf{Q}}^2 \quad (2)$$

where we use the notation $\| \mathbf{x} \|_{\mathbf{Q}}^2$ to represent the quadratic form $\mathbf{x}^T \mathbf{Q} \mathbf{x}$, and \mathbf{Q} is a symmetric, positive semi-definite weighting matrix. It is easy to see that when \mathbf{Q} is an identity matrix (i.e. minimizing Euclidean distance in Equation 1), the weighted and standard LK objective functions are equivalent. For this reason we shall refer to the standard LK algorithm herein as *Euclidean LK*.

Minimizing the error in Equation 2 is a non-linear optimization task. The weighted LK algorithm is able to find an effective solution to Equation 2, by iteratively linearizing $I(\mathbf{p} + \Delta\mathbf{p})$ and refining the initial guess $\mathbf{p} \leftarrow \mathbf{p} + \Delta\mathbf{p}$ at each iteration. The linearized objective function now takes

the form: $E \approx \| I(\mathbf{p}) + \mathbf{J}\Delta\mathbf{p} - T(\mathbf{0}) \|_{\mathbf{Q}}^2$, where \mathbf{J} is the Jacobian of $I(\mathbf{p})$ defined by: $\mathbf{J} = \left(\frac{\partial I(\mathbf{p})}{\partial \mathbf{p}} \right)^T$. The explicit solution of $\Delta\mathbf{p}$ that minimizes the linearized objective function is,

$$\Delta\mathbf{p} = \mathbf{H}^{-1}\mathbf{J}^T\mathbf{Q}[T(\mathbf{0}) - I(\mathbf{p})] \quad (3)$$

where the pseudo Hessian matrix is defined by $\mathbf{H} = \mathbf{J}^T\mathbf{Q}\mathbf{J}$. The algorithm proceeds to update current warp parameters ($\mathbf{p} \leftarrow \mathbf{p} + \Delta\mathbf{p}$) iteratively till convergence.

A central motivation for using the weighted LK algorithm, over the Euclidean LK algorithm, is the realization that prior knowledge of the salience and correlation of pixels can be naturally encoded into \mathbf{Q} leading to more robust performance during alignment/tracking in real-world scenarios. An inherent problem, however, with the weighted LK algorithm lies in choosing an appropriate weighting matrix \mathbf{Q} . For computational reasons the expressiveness of \mathbf{Q} has largely been restricted to low rank or diagonal matrices severely limiting the robustness and usefulness of the weighted LK algorithm.

4. LK on banks of filter responses

Linear filters are often used to extract useful feature representations in computer vision. One particular filter, based on the seminal work of Gabor [13], that has received much attention in the vision community are Gabor wavelets due to their biological relevance and computational properties [10, 9, 11, 12]. The employment of a concatenation of Gabor filter responses, as a pre-processing step to deal with illumination change, before learning a classifier has found particular success in face identity [24, 18] and expression [6] recognition when compared to learning those classifiers with original appearance features/pixels. For a review on Gabor filters see Appendix A.

In this paper we want to explore the application of this idea to image registration within the LK framework. We formulate the LK alignment problem for images represented in the overcomplete Gabor response domain. If \mathbf{g}_i is the impulse response of the i -th Gabor filter and there are M filters in total, then we are interested in the warp that minimizes the sum of squared differences (SSD) simultaneously across all filter responses of the warped image and the template image. This error may be written as,

$$E = \| \{ \mathbf{g}_i * I(\mathbf{p}) \}_{i=1}^M - \{ \mathbf{g}_i * T(\mathbf{0}) \}_{i=1}^M \|_2^2 \quad (4)$$

where $*$ represents a 2D convolution operation while $\{ \cdot \}_{i=1}^M$ represents the concatenation operation i.e. $\{ \mathbf{x}_i \}_{i=1}^M = [\mathbf{x}_1^T \dots \mathbf{x}_M^T]^T$. Unsurprisingly, the form of the above equation is very similar to that of standard LK (Equation 1). However, there are a number of issues with this naïve extension.

Computational concerns: As pointed out by [18, 6, 5] a particular problem with this approach is the inherently large memory and computational overheads required for representing images in this over-complete Gabor domain. Applying this strategy to the LK framework presents two fundamental problems. First, if there are M filters in the bank, and N pixels in the input image, we need to do M 2D convolutions involving images containing N pixels each. Second, the size of the Jacobian \mathbf{J} (in Equation 3) for an affine warp described by n warp parameters, becomes $MN \times n$ instead of being $N \times n$. As a result of these computational overheads, the idea of doing LK alignment with even a modest number of Gabor filters in the bank (e.g., 9 scales times 8 orientations, i.e. $M = 72$, as employed in [18]) becomes prohibitively expensive and impractical.

Even for smaller filter bank sizes authors in literature have resorted to methods for approximating the full response vectors such as: (i) downsampling of filter responses [18], (ii) employing filter responses at certain fiducial positions within the image [24], (iii) the employment of feature selection methods to select the most discriminative filter responses [6], and most recently (iv) where individual classifiers are learnt for each filter response and a fusion strategy employed to combine the outputs in a synergistic manner [17].

In the following section we demonstrate that by casting the LK algorithm in the Fourier domain, we can eliminate the need for any of the above approximations and derive a computationally efficient LK algorithm that can fit simultaneously across multiple filter bank responses. Further, analysis of this approach allows us to also answer some fundamental questions about the selection and expressiveness of the weighting matrix \mathbf{Q} associated with the weighted LK algorithm.

5. Fourier LK

It is elementary to show that the error in Equation 4 can equivalently be written as,

$$E = \sum_{i=1}^M \| \mathbf{g}_i * [I(\mathbf{p}) - T(\mathbf{0})] \|_2^2 \quad (5)$$

Exploiting the fact that convolution becomes a Hadamard (i.e., element-by-element) product in the Fourier domain, and employing Parseval's relation [21] (energy content is preserved as we move from the spatial to the Fourier domain), we may write the error in Equation 5 as follows,

$$E = \| \hat{I}(\mathbf{p}) - \hat{T}(\mathbf{0}) \|_{\mathbf{S}}^2 \quad (6)$$

where,

$$\mathbf{S} = \sum_{i=1}^M (\text{diag}(\hat{\mathbf{g}}_i))^T \text{diag}(\hat{\mathbf{g}}_i) \quad (7)$$

and $\hat{I}, \hat{T}, \hat{\mathbf{g}}_i$ are the 2D Fourier transforms of vectorized I, T, \mathbf{g}_i respectively. We should also note that any transpose operator T on a complex vector or matrix in this paper additionally takes the complex conjugate in a similar fashion to the Hermitian adjoint [21]. The matrix \mathbf{S} is a *diagonal* matrix that can be *precomputed* and is *independent* of the number of filters being applied. We also know that the operation of a 2D Fourier transform can be replaced by pre-multiplying a signal (of length N) by a $N \times N$ matrix \mathbf{F} containing the Fourier basis vectors. This can be seen in the following FLK objective function,

$$E = \| I(\mathbf{p}) - T(\mathbf{0}) \|_{\mathbf{F}^T \mathbf{S} \mathbf{F}}^2 \quad (8)$$

A link to weighted LK: By casting the LK algorithm in the Fourier domain, we have shown that doing alignment on linear filter preprocessed images is equivalent to the weighted LK algorithm with a weighting matrix $\mathbf{Q} = \mathbf{F}^T \mathbf{S} \mathbf{F}$ where \mathbf{S} (Equation 7) is determined by the choice of filters being used. Deriving the weighting matrix \mathbf{Q} directly from a bank of Gabor filters is advantageous as they naturally encode redundancies present in the human visual system. Moreover we have described a practical method for applying a richer set of weighting matrices ($\mathbf{Q} = \mathbf{F}^T \mathbf{S} \mathbf{F}$ is full rank and non-diagonal) in the spatial domain by just applying a diagonal weighting matrix \mathbf{S} in the Fourier domain. Inspecting Equation 8 one can see that for the case when \mathbf{S} is an identity matrix the *Euclidean FLK* algorithm is equivalent to the traditional Euclidean LK algorithm due to Parseval’s relation. From herein we shall refer to the FLK algorithm employing Gabor filters as the *Gabor FLK* algorithm.

Computational concerns: Theoretically, one should be able to apply the weighted LK algorithm, described in Section 3, blindly with our newly derived Fourier weighting matrix in Equation 7. However, the newly derived matrix \mathbf{Q} in the spatial domain is full rank (although inherently sparse) so we would face the same computational concerns faced by any full rank matrix irrespective of whether it is derived in the Fourier domain or not. What makes this approach computationally feasible is that we can replace the matrix form of the Fourier transform \mathbf{F} (which has a cost of $O(N^2)$) with a computationally feasible Fourier transform that is $O(N \log N)$ [21]. This can be seen more clearly if one inspects the weighted FLK warp update equation,

$$\Delta \mathbf{p} = \mathbf{H}_{flk}^{-1} (\mathbf{F} \mathbf{J})^T \mathbf{S} \mathbf{F} [T(\mathbf{0}) - I(\mathbf{p})] \quad (9)$$

where a Fourier transform only needs to be applied to the columns of the Jacobian matrix \mathbf{J} and the error image $T(\mathbf{0}) - I(\mathbf{p})$ at each iteration. The pseudo-Hessian can then be computed $\mathbf{H}_{flk} = (\mathbf{F} \mathbf{J})^T \mathbf{S} (\mathbf{F} \mathbf{J})$. A full breakdown of the computational cost of the FLK algorithm can be seen in Table 1 (a).

6. Experiments

We conducted our experiments, using a similar methodology as [4], as follows. Given an experimental image of an object (face, cup, car, etc.) we manually selected a template T centered around the object of interest. We then randomly generated affine warps as follows. We selected three canonical points indicated by red pluses in Figure 1(a). We then randomly perturbed these three points with additive white Gaussian noise of certain variance. The new location of these points describe an affine warp. Sample affine warps generated using this method are shown in Figure 1(b). We then fit for the affine warp parameters defined by these three points. To account for the difference of units amongst the six affine parameters, we use the following error measure rather than the error in the parameters [4]. Given the current estimate of the warp, we compute the destination of the three canonical points and compare them with their correct locations in terms of the RMS error between their current and correct locations. For the warps shown in Figure 1(b), the RMS error between the warped canonical points and their correct locations (Figure 1(a)) is ~ 30 pixels.

To compare the algorithms in terms of their robustness to illumination variability, we explored the case when the lighting conditions of the input image were very different from that of the template image (Figure 1(c)). As an illustration of the quality of fitting performed by the two algorithms in the presence of illumination variability, we show a comparison in Figure 2. In Figure 2(b) the Euclidean FLK is off target, while in Figure 2(c) Gabor FLK shows its robustness to illumination changes. For a quantitative assessment we compared the two algorithms in terms of average convergence rates and average convergence frequency as described in the following sections. For this quantitative assessment we employed the Multi-PIE face image dataset [15] with hand annotated ground-truth of the eye and nose positions.

Average convergence rates: To generalize the results of previous section, we conducted fitting experiments on a large number (3000) of randomly generated warps. The initial RMS error for these warps ranged from 10-35 pixels. The RMS error plotted as a function of the iteration number is shown in Figure 3. When the lighting conditions for the template and the input image are the same, both algorithms have similar performance (Figure 3(a)). When the lighting conditions for the input image are changed, the Gabor FLK method is robust to the variability, while the Euclidean FLK algorithm diverges (Figure 3(b)).

Average frequency of convergence: To check the average frequency of convergence given an initial RMS error, we generated 3000 warps (500 warps per initial RMS error) ranging from 10-35 pixels. The algorithm was reckoned to be converged if the final error was less than 5 pixels. We present the curves for convergence frequency, as a percent-



Figure 1. (a) A template centered around the object of interest with three canonical points shown as red pluses. (b) Sample warps defined by the random perturbation of the three canonical points. (c) Sample warps shown on an image that has different illumination conditions as compared to that of the template for various objects.

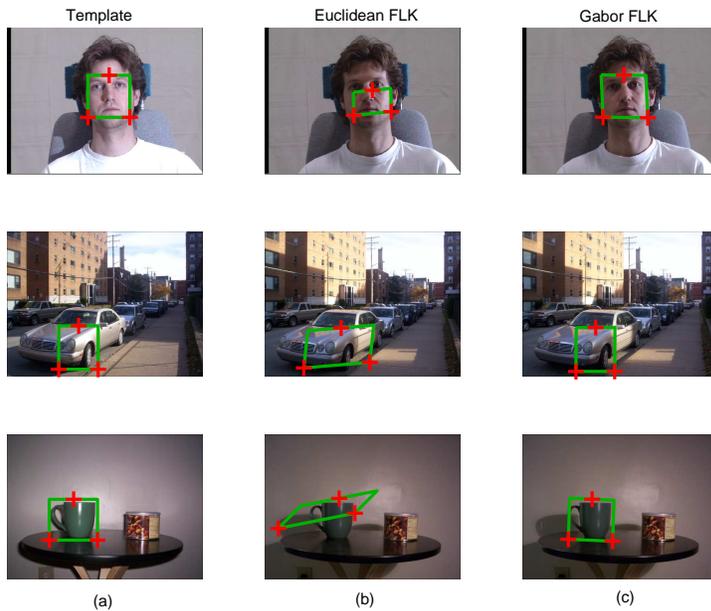


Figure 2. Example fittings for the warps shown in Figure 1 in the presence of illumination. (a) Shows the template image. (b) Euclidean FLK is off-target in the presence of illumination variability in the source image. (c) Gabor FLK is robust in the presence of illumination variability in the source image. (** The reviewers are encouraged to view the video accompanying this submission)

age, in Figure 4. Again the two algorithms have similar performance when the lighting conditions are the same for the input and the template image (Figure 4(a)). For the case when the input image has a different illumination, the Gabor FLK clearly outperforms the Euclidean FLK whose convergence frequency remains virtually zero for all the values of

initial RMS error (Figure 4(b)). This clearly demonstrates the hypersensitivity of the conventional LK to illumination changes, while underscoring the robustness of Gabor FLK algorithm.

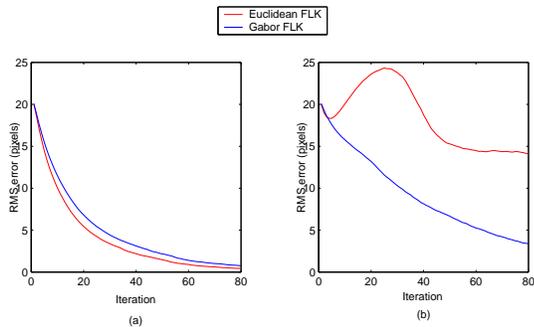


Figure 3. Average convergence rates over 3000 randomly generated warps ranging from an initial RMS error of 10-35 pixels. (a) When the input and template images have the same illumination conditions, both algorithms perform equally well. (b) When the illumination of the input image changes, the Gabor FLK algorithm is still able to do the fitting, while the Euclidean FLK fails.

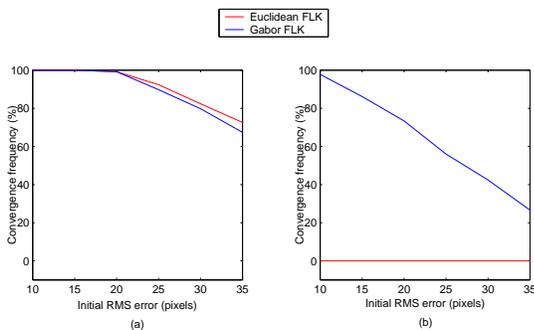


Figure 4. Average frequency of convergence as a function of initial RMS. (a) When the input and template images have the same illumination conditions, both algorithms perform equally well. (b) When the illumination of the input image changes, the Gabor FLK algorithm clearly outperforms the Euclidean FLK for which the convergence frequency remains zero for all values of initial RMS error.

7. Inverse compositional FLK

The FLK formulation presented in Section 5 adheres to the canonical form of the LK algorithm known as the forwards additive (FA) algorithm [4]. A fundamental problem with the forwards additive approach is that it requires the re-estimation of the Hessian matrix at each iteration greatly impacting computational efficiency. Notably, Baker and Matthews [3] presented a computationally efficient extension to forwards additive LK which they refer to as the inverse compositional (IC) algorithm. Like the forwards additive algorithm the goal is to minimize the SSD objective function described in Equation 1. The approach differs, however, in that it linearizes $T(\Delta\mathbf{p})$ rather than $I(\mathbf{p} + \Delta\mathbf{p})$ resulting in the following linearized objective function,

$$E \approx \| T(\mathbf{0}) + \mathbf{J}_{ic}\Delta\mathbf{p} - I(\mathbf{p}) \|^2 \quad (10)$$

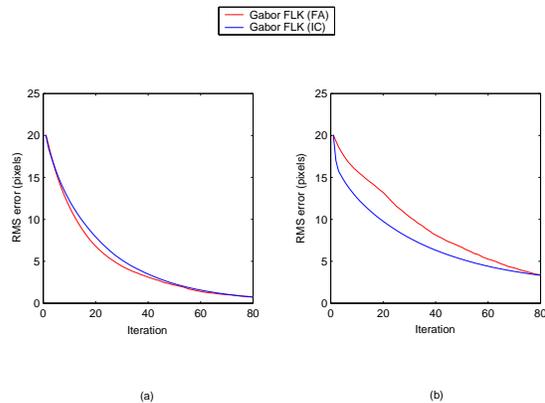


Figure 5. Equivalence between forwards additive and inverse compositional for average convergence rates over 3000 randomly generated warps ranging from an initial RMS error of 10-35 pixels (a) when the input and template images have the same illumination conditions, and (b) when the input and template images have different illumination conditions.

where $\mathbf{J}_{ic} = \left(\frac{\partial T(\mathbf{0})}{\partial \mathbf{p}} \right)^T$. One can then solve this linearized objective function,

$$\Delta\mathbf{p} = \mathbf{B} [I(\mathbf{p}) - T(\mathbf{0})] \quad (11)$$

where \mathbf{B} can be completely pre-computed and the current warp parameters are iteratively updated by the inverse (as we want to update the source image not the template) of the warp update $\mathbf{p} \leftarrow \mathbf{p} \circ \Delta\mathbf{p}^{-1}$. The operation \circ represents the composition of two warps which, in the case of an affine warp, can easily be represented as a matrix multiplication.

Estimating \mathbf{B} for FLK: For the specific case of the FLK objective function defined in Equation 5,

$$\mathbf{B} = \mathbf{H}_{flk(ic)}^{-1} (\mathbf{F}\mathbf{J}_{ic})^T \mathbf{S}\mathbf{F} \quad (12)$$

As with all instances of the inverse compositional approach the Jacobian \mathbf{J}_{ic} depends only on the template image i.e.

$\mathbf{J}_{ic} = \left(\frac{\partial T(\mathbf{0})}{\partial \mathbf{p}} \right)^T$ which remains constant across all iterations. Consequently, the pseudo-Hessian $\mathbf{H}_{flk(ic)} = \mathbf{J}_{ic}^T \mathbf{F}^T \mathbf{S}\mathbf{F} \mathbf{J}_{ic}$ also remains static for all the iterations.

The computational dividends accrued by the inverse compositional formulation of the FLK algorithm are evident in Table 1. If n and N represent the number of warp parameters and the number of pixels in the template image respectively, the per-iteration computational complexity of forwards additive FLK is $O(n^3 + n^2N + nN \log N)$, while that of inverse compositional FLK is $O(n^2 + nN)$. We would like to emphasize that in both the cases, the complexity is independent of the number of filters, M . The empirical equivalence between the two formulations is demonstrated in Figures 5 and 6.

No.	Step	Complexity
1	Warp I with \mathbf{p} to compute $I(\mathbf{p})$	$O(nN)$
2	Compute the error image: $I(\mathbf{p}) - T(\mathbf{0})$	$O(N)$
3	Evaluate the Jacobian $\mathbf{J} = \left(\frac{\partial I(\mathbf{p})}{\partial \mathbf{p}}\right)^T$	$O(nN)$
4	Compute FFT of the Jacobian	$O(nN \log N)$
5	Compute the Hessian \mathbf{H}_{flk}	$O(n^2 N + nN)$
6	Invert the Hessian \mathbf{H}_{flk}	$O(n^3)$
7	Compute FFT of the error image	$O(N \log N)$
8	Compute $(\mathbf{F}\mathbf{J})^T \mathbf{S}\mathbf{F}[I(\mathbf{p}) - T(\mathbf{0})]$	$O(nN + N)$
9	Compute $\Delta \mathbf{p}$	$O(n^2)$
10	Update \mathbf{p} : $\mathbf{p} \leftarrow \mathbf{p} + \Delta \mathbf{p}$	$O(n)$
	Total	$O(n^3 + n^2 N + nN \log N)$

No.	Step	Complexity
1	Evaluate the Jacobian \mathbf{J}_{ic}	$O(nN)$
2	Compute FFT for the Jacobian	$O(nN \log N)$
3	Compute the Hessian $\mathbf{H}_{flk(ic)}$	$O(n^2 N + nN)$
4	Invert the Hessian $\mathbf{H}_{flk(ic)}$	$O(n^3)$
5	Compute the matrix \mathbf{B} (Eq. 12)	$O(n^2 N + nN + nN^2)$
	Total	$O(n^3 + n^2 N + nN^2 + nN \log N)$

No.	Step	Complexity
1	Warp I with \mathbf{p} to compute $I(\mathbf{p})$	$O(nN)$
2	Compute the error image: $I(\mathbf{p}) - T(\mathbf{0})$	$O(N)$
3	Compute $\Delta \mathbf{p} = \mathbf{B}[I(\mathbf{p}) - T(\mathbf{0})]$	$O(nN)$
4	Update \mathbf{p} : $\mathbf{p} \leftarrow \mathbf{p} \circ \Delta \mathbf{p}^{-1}$	$O(n^2)$
	Total	$O(n^2 + nN)$

Table 1. Comparison of per-iteration complexity between the forwards additive and the inverse compositional formulations of the FLK algorithm. n and N represent the number of warp parameters and the number of pixels in the template image respectively, (a) per-iteration complexity for the FA algorithm, (b) complexity of pre-computations for the IC algorithm, and (c) per-iteration complexity for the IC algorithm.

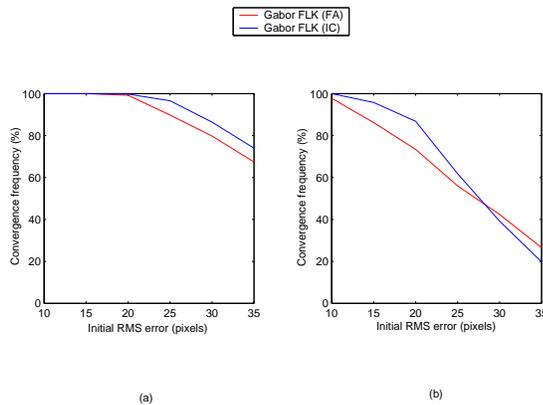


Figure 6. Equivalence of forwards additive (FA) and inverse compositional (IC) for average frequency of convergence as a function of initial RMS, (a) When the input and template images have the same illumination conditions, and (b) When the input and template images have different illumination conditions.

8. Discussion

The results in Figures 2, 3 and 4 show that the Gabor FLK algorithm clearly outperforms the conventional Euclidean FLK. A natural question arising from this result is: “Why does fitting across multiple filtered image responses improve alignment performance?” We can gain interesting insights to this question in light of the equivalence we established in this paper between the application of such filters and the manipulation of the distance metric used to quantify the alignment mismatch. A message that stems out of our current work is that the choice of the distance metric is crucial to the performance of an alignment algorithm, and linear filters provide a principled method to manipulate the distance metric through the weighting matrix \mathbf{S} . This result is consistent with parallel messages coming out of some recent works in the machine learning community [22]. A

more interesting question should perhaps be now, “What is the best \mathbf{S} for my alignment problem?” rather than “What filter banks give best performance?”

9. Conclusions

In this paper we have demonstrated that the LK algorithm can be equivalently cast in the Fourier domain. This formulation allows us to interpret the simultaneous alignment across multiple filter responses as a form of the weighted LK algorithm. This interpretation in turn enables us to use a much richer class of computationally practical weighting matrices (full-rank, non-diagonal) for the weighted LK algorithm. Hitherto, only simpler matrices (low rank, diagonal) were explored as weighting matrices [2]. Further, we have presented a method to do image alignment on the bank of image responses derived from Gabor filters. We have shown that doing image alignment in the high dimensional Gabor feature space is mathematically equivalent to doing alignment in the low dimensional image space, if appropriate weightings are applied in the Fourier domain. We have compared our alignment algorithm (Gabor FLK) with the conventional Euclidean FLK and have shown that Gabor FLK outperforms Euclidean FLK in the presence of variabilities like illumination changes. Most notably, we have demonstrated how the computationally efficient inverse compositional framework can be applied to the FLK algorithm for fast, robust (particularly to image fluctuations), and accurate image alignment.

A. Gabor Feature Representations

In the 2D spatial domain, a Gabor wavelet is a complex exponential modulated by a Gaussian,

$$g_{\omega,\theta}(x, y) = \frac{1}{2\pi\sigma^2} \exp\left\{-\frac{x'^2 + y'^2}{2\sigma^2} + j\omega x'\right\} \quad (13)$$

where $x' = x \cos(\theta) + y \sin(\theta)$, $y' = -x \sin(\theta) + y \cos(\theta)$, x and y denote the pixel positions, ω represents the centre frequency, θ represents the orientation of the Gabor wavelet, while σ denotes the standard deviation of the Gaussian function. Please refer to [12] on strategies for spacing the filters in the 2D spatial frequency domain for a fixed number of scales and orientations. Given a N dimensional vectorized input image \mathbf{x} and the vectorized $P \times Q$ 2D filter $\mathbf{g}_{\omega,\theta} = [g_{\omega,\theta}(1,1), \dots, g_{\omega,\theta}(P,Q)]^T$ one can obtain the N dimensional vector response,

$$\hat{\mathbf{r}}_{\omega,\theta} = \hat{\mathbf{x}} \circ \hat{\mathbf{g}}_{\omega,\theta} \quad (14)$$

where $\hat{\mathbf{g}}_{\omega,\theta}$, $\hat{\mathbf{x}}$ and $\hat{\mathbf{r}}_{\omega,\theta}$ ² are the vectorized complex 2D discrete Fourier transforms (DFT) [21] of the vectorized real images $\mathbf{g}_{\omega,\theta}$, \mathbf{x} and $\mathbf{r}_{\omega,\theta}$ respectively, while \circ represents the operation of taking the Hadamard product between two vectors. In the common case where $N > PQ$, $\mathbf{g}_{\omega,\theta}$ can be padded with $N - PQ$ zeros to ensure it is the same size as \mathbf{x} . We should note that the operation in Equation 14 can be equivalently accomplished purely in the image (spatial) domain through the use of efficient 2D convolution operators, however, we have chosen to employ a Fourier representation in this paper due to its particularly useful ability to represent 2D convolution as a Hadamard product in the Fourier domain.

References

[1] S. Avidan. Support vector tracking. *IEEE Trans. PAMI*, 26(8):1064–1072, August 2004. 2

[2] S. Baker, R. Gross, I. Matthews, and T. Ishikawa. Lucas-kanade 20 years on: A unifying framework: Part 2. Technical Report CMU-RI-TR-03-01, Robotics Institute, Pittsburgh, PA, February 2003. 1, 2, 7

[3] S. Baker and I. Matthews. Equivalence and efficiency of image alignment algorithms. In *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2001. 1, 6

[4] S. Baker and I. Matthews. Lucas-kanade 20 years on: A unifying framework: Part 1: The quantity approximated, the warp update rule, and the gradient descent approximation. *International Journal of Computer Vision*, 56(3):221–255, February 2004. 1, 2, 4, 6

[5] M. Bartlett, G. Littlewort, C. Lainscsek, I. Fasel, M. Frank, and J. Movellan. Fully automatic facial action recognition in spontaneous behavior. *7th International Conference on Automatic Face and Gesture Recognition*, 2006. 3

[6] M. S. Bartlett, G. Littlewort, M. Frank, C. Lainscsek, I. Fasel, and J. Movellan. Recognizing facial expression: Machine learning and application to spontaneous behavior. *IEEE Conference on Computer Vision and Pattern Recognition*, 2:568–573, June 2005. 2, 3

²Please note that throughout this paper that the notation $\hat{\cdot}$ applied to any vector denotes the 2D-DFT of a vectorized 2D image such that $\hat{\mathbf{x}} \leftarrow \mathbf{F}\mathbf{x}$, where \mathbf{F} is the $N \times N$ matrix of complex basis vectors for mapping to the 2D Fourier domain for any N dimensional vectorized image.

[7] A. Bartoli. Groupwise geometric and photometric direct image registration. *IEEE Trans. PAMI*, 30(12):2098–2108, December 2008. 2

[8] M. Black and A. Jepson. Eigen-tracking: Robust matching and tracking of articulated objects using a view-based representation. *International Journal of Computer Vision*, 36(2):101–130, 1998. 1, 2

[9] J. Daugman. Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two-dimensional cortical filters. *Journal of the Optical Society of America*, 2(7):1160–1169, 1985. 3

[10] J. G. Daugman. Two-dimensional spectral analysis of cortical receptive field profiles. *Vision Research*, 20(10):847–856, 1980. 3

[11] J. G. Daugman. Complete discrete 2-D Gabor transforms by neural networks for image analysis and compression. *IEEE Trans. PAMI*, 36:1169–1179, July 1988. 3

[12] D. J. Field. Relations between the statistics of natural images and the response properties of cortical cells. *Journal of the Optical Society of America A*, 4(12):2379–2393, 1987. 3, 8

[13] D. Gabor. Theory of communication. *Journal of the Institution of Electrical Engineers (London)*, 93(III):429–457, 1946. 3

[14] R. Gross, S. Baker, and I. Matthews. Generic vs. person specific active appearance models. *Image and Vision Computing*, 23(11):1080–1093, November 2005. 2

[15] R. Gross, J. S. Baker, I. Matthews, and T. Kanade. Multi-PIE. In *IEEE International Conference on Automatic Face and Gesture Recognition*, 2008. 4

[16] G. D. Hager and P. N. Belhumeur. Efficient region tracking with parametric models of geometry and illumination. *IEEE Trans. PAMI*, 20:1025–1039, 1998. 1, 2

[17] Z. Li, D. Lin, and X. Tang. Nonparametric discriminant analysis for face recognition. *IEEE Trans. PAMI*, 31(4):755–761, April 2009. 2, 3

[18] C. Liu and H. Wechsler. Gabor feature based classification using the enhanced Fisher linear discriminant model for face recognition. *IEEE Trans. Image Processing*, 11(4):467–476, 2002. 2, 3

[19] B. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *International Joint Conference on Artificial Intelligence*, pages 674 – 679, 1981. 1, 2

[20] S. Lucey. Enforcing non-positive weights for stable support vector tracking. In *IEEE International Conference on Computer Vision and Pattern Recognition*, June 2008. 2

[21] A. V. Oppenheim and A. S. Willsky. *Signals & Systems*. Prentice Hall, 2nd edition, 1996. 2, 3, 4, 8

[22] P. K. Shivaswamy and T. Jebara. Relative margin machines. In *Neural Information Processing Systems 21 (NIPS)*, 2008. 7

[23] B. Theobald. Evaluating error functions for robust active appearance models. In *IEEE International Conference on Automatic Face and Gesture Recognition*, pages 149–154, 2006. 2

[24] C. Wiskott, J. M. Fellous, N. Krüger, and C. von der Malsburg. Face recognition by elastic bunch graph matching. *IEEE Trans. PAMI*, 19(7):775–779, July 1997. 2, 3