

Finite Element Image Warping

Peter Kaufmann¹ Oliver Wang¹ Alexander Sorkine-Hornung¹ Olga Sorkine-Hornung² Aljoscha Smolic¹ Markus Gross^{1,2}

¹Disney Research Zurich ²ETH Zurich



Figure 1: Our unifying representation of image warping supports efficient adaptive meshing, high order basis functions, and more. a) Original image, b) automatic saliency map, c) uniform scaling, d) retargeting using existing methods (6767 degrees of freedom (DOF)), e) retargeting using an adaptive mesh supported by our FEM framework (1325 DOF).

Abstract

We introduce a single unifying framework for a wide range of content-aware image warping tasks using a finite element method (FEM). Existing approaches commonly define error terms over vertex finite differences and can be expressed as a special case of our general FEM model. In this work, we exploit the full generality of FEMs, gaining important advantages over prior methods. These advantages include arbitrary mesh connectivity allowing for adaptive meshing and efficient large-scale solutions, a well-defined continuous problem formulation that enables clear analysis of existing warping error functions and allows us to propose improved ones, and higher order basis functions that allow for smoother warps with fewer degrees of freedom. To support per-element basis functions of varying degree and complex mesh connectivity with hanging nodes, we also introduce a novel use of discontinuous Galerkin FEM. We demonstrate the utility of our method by showing examples in video retargeting and camera stabilization applications, and compare our results with previous state of the art methods.

1. Introduction

Content-aware image warping has recently been shown to be a powerful tool in a wide range of editing applications. Such methods modify images by overlaying a mesh and solving for an optimal, locally-varying deformation that minimizes some application-specific set of constraints. In traditional solutions, the constraints are defined in terms of vertex finite differences computed on a regular grid, or by discretizing the image into a quad mesh and computing per-quad energies from the distortion of grid edges. The error function is then minimized, generally by formulating it as a large sparse system of equations.

While finite difference based metrics such as these are a straightforward representation of pixel-based image data, they tightly couple error terms with the mesh structure, making it difficult to extend the problem formulation into new domains. Instead, we introduce a *unifying* representation for a wide range of image editing tasks by using a finite element method (FEM) that includes existing finite difference metrics as a *special case*. Our approach constructs a single robust mathematical formulation of the general *continuous* image warping problem, and allows us to leverage deformation knowledge from mechanics and geometry communities.

Since its invention over half a century ago, the finite element method has become one of the most popular means for solv-

ing partial differential equations. Its main strength over alternative methods is that it supports irregular discretizations of the problem domain (resulting in better approximations of the boundary geometry) and allows refinement of the discretization in important areas of the domain. The FEM finds wide application in computer graphics, predominantly in the field of physically-based simulation where it is used for the realistic simulation of deformable objects or fluids. However, except for a few cases, this method has been largely ignored in the image processing domain.

The FEM owes its success to a rigorous mathematical foundation; for certain types of problems, error estimates and convergence properties can be derived, justifying a high confidence in the computed results. Moreover, assuming certain bounds on element size, the computed solutions are largely independent of the actual element mesh. These facts justify the use of locally adaptive meshes, which is an extremely important requirement for large scale solutions (enabling stable video warping), but one that has been largely ignored due to difficulties in formulation.

Furthermore, given our continuous framework, it becomes easier to validate and justify the use of specific energy functions that drive the warping. We discuss how existing energy functions can be phrased in a continuous sense, and propose simple novel energy functions with added benefits, such as effective prevention of warp inversions (a significant issue with finite-difference based approaches) without resorting to nonlinear workarounds or incorporating inequality constraints that require quadratic programming to solve.

Another added benefit of our general formulation is that our approach allows for higher order basis functions at element vertices. Unlike prior work that simply averages per-quad information, this allows us to sample high-resolution image information within elements, while still performing a low degree of freedom minimization.

We present a novel discontinuous Galerkin FEM (DG FEM) formulation that allows working with meshes of arbitrary connectivity, with support for hanging nodes (edge nodes that do not belong to all elements that share the edge) that traditional FEMs cannot support.

We demonstrate our FEM image warping in the domain of media retargeting, where the aspect ratio of images or videos is modified such that the shape of important content is preserved, as well as video stabilization, where a smooth reconstruction of hand-held shaky videos is computed. We show adaptive solutions that allow us to solve a whole sequence of frames at once; a task that previously would be either computationally intractable, or have only low resolution control due to excessive subsampling.

To summarize, our contribution is a novel, general representation for continuous locally-varying image warping that models deformation using an FEM. We enumerate advantages of this approach, and demonstrate them in high-quality,

temporally consistent video resizing and video stabilization applications.

2. Related Work

The FEM has a long history in computer graphics. It has been commonly used in domains such as physically based animation [TF88], and remains the method of choice for the simulation of deformable objects. Recent applications include geometric modeling [JTSZ10] and surface parameterization [SLS*07]. One can distinguish between linear FEM where the corresponding error function is quadratic in the unknowns and its minimum can be found by solving a sparse linear system, and the more general case of nonlinear FEM leading to a nonlinear minimization problem [BW97]. Next to the “standard textbook” FEM, a number of variants exist, including discontinuous Galerkin FEM (DG FEM), mixed FEM, and extended FEM (X-FEM). An FEM based image warping framework opens up all of these possibilities, and we make particular use of DG FEM [Coc03] which, thanks to its less strict continuity requirements, allows for combining elements in ways that would not be possible with standard FEM. DG FEM has already found applications in computer graphics [KMBG08], where its support for arbitrary non-convex polyhedral elements allows for the efficient simulation of deformable object cutting. We are mainly interested in DG FEM because it allows for easy meshing and combining elements of different polynomial degree.

These successes motivate our use of the FEM in the image warping domain, where with the exception of a few methods it has largely been ignored. One such method proposes the use of finite elements in medical image warping for registration [Gee94]. However, in this case, a simple linear finite element model is used. We provide a higher order model, and take advantage of numerous other benefits of the FEM for our warping solution.

Traditional image-based warping is on the other hand, a long running and large area of research within computer graphics. Beier et al. [BN92] present a classic example of mesh-based image warping that morphs between images by mapping features. More recently, advances in computing power have allowed for content-aware image warping techniques that compute globally optimal distortions of images. These methods have been successful in a wide range of applications, such as: media retargeting [SS09], video stabilization [LGJA09], fish-eye lens distortion correction [CAA09], perspective modification [CAA10], and stereoscopic editing [LHW*10]. Our novel representation of image warping encompasses all of these solutions and gains advantages from the more generalized formulation, such as support for adaptive meshes, higher order basis functions, temporal stability and a continuous formulation. We will address each of these advantages in the context of related work.

Support for adaptive meshes of arbitrary connectivity is a significant benefit of our FEM approach and is required for high-quality temporally consistent results. Previous methods that compute solutions for full video sequences have had to choose between two options: representing videos with a sparsely sampled mesh, which gives insufficient control over regions that require high-frequency changes in distortion [WLSL10], or using a dense representation, which quickly scales beyond reasonable computation for video sequences (For 1080p HD resolution, this results in about 2 million free variables per frame. Given a short sequence of 200 frames, that leads to roughly 400 million free variables). As a result, many methods have attempted to reduce the effects of temporal artifacts while solving for local deformations by enforcing neighboring frame consistency on a frame-by-frame or windowed basis [GWCO09, KLHG09], or by introducing motion-aware importance maps [WFS*09, NLLG10]. Recent work reduces the complexity of the problem by only allowing for axis-aligned warps [PWS12], which however can be too restrictive, especially when the image contains larger areas of homogeneous color. In contrast to previous methods, our method can solve a full sequence of frames at once without sacrificing accuracy. This is possible thanks to an adaptive FEM mesh, which substantially reduces the total number of degrees of freedom of the problem without harming visual quality.

Meshes with multiple levels of refinement have been used for a content-aware zooming application [LJW*10]. In this case, a Delaunay triangulation creates an initial mesh with denser mesh levels created by triangle subdivision. However, this method can only use a combination of several fixed-resolution meshes. We express our error formulation *independently* of mesh connectivity, which lets us use any mesh subdivision technique.

Furthermore, our general representation also allows for higher order basis functions at element vertices. This is in contrast to prior finite difference image warping methods, which use a piecewise-linear approximation, resulting in an averaging of high-resolution image content within mesh elements.

3. FEM for Image Warping

In order to derive an FEM for image warping, we begin by formulating the general image warping problem in the continuous case, and then in Section 4 we will discuss application specific decisions. While previous approaches first set up the discretization using finite differences, regularly spaced grids, or triangle meshes, computing ad-hoc energy functions from the resulting primitives (vertices, edges), a continuous formulation allows us to study and compare the properties of various image warping energies independently of their discretization. We *then* perform the actual discretization by means of finite elements as a second, independent

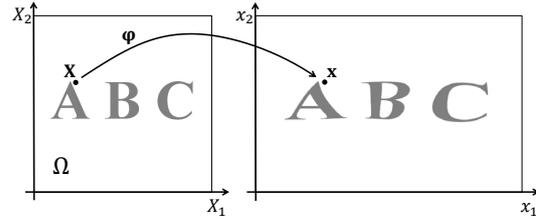


Figure 2: Continuous mapping from the undeformed domain to the warped image using a simple nonuniform scale warp.

step, where we are presented with a multitude of choices, allowing us for example to trade accuracy (how well the continuous solution is approximated) for performance.

3.1. Continuous Warping

Consider the rectangular domain $\Omega = [0, 1] \times [0, h] \subset \mathbb{R}^2$ of an undeformed image. The warping function $\varphi : \Omega \rightarrow \mathbb{R}^2$ maps a point $X = (X_1, X_2)^T \in \Omega$ to a warped point $x = (x_1, x_2)^T = (\varphi_1(X_1, X_2), \varphi_2(X_1, X_2))^T = \varphi(X)$ (see Figure 2). Following the notation used in the mechanics community, we define the *deformation gradient* as the 2×2 matrix $F(X)$ with entries:

$$F_{ij}(X) = \frac{\partial \varphi_i}{\partial X_j} \Big|_X$$

In order to define the cost of performing a certain warp φ , we introduce a function Ψ that computes the *deformation energy density* (per undeformed area) at any point $X \in \Omega$. The total deformation energy of a warp φ can then be computed as:

$$E[\varphi] = \int_{\Omega} \Psi(F(X)) dX$$

The optimal warp is the one that minimizes E , and respects a number of problem-specific boundary constraints defined in Section 4.

3.2. FEM Discretization

Now that we have defined the continuous image warping problem, we discretize the problem using the FEM in order to numerically compute the optimal warp.

Basis Functions As a first step, the warping function φ is discretized into a linear combination of n *basis functions* $N^a : \Omega \rightarrow \mathbb{R}$ with associated weights $x^a = (x_1^a, x_2^a)^T \in \mathbb{R}^2$:

$$\varphi(X) = \sum_{a=1}^n N^a(X) x^a \quad (1)$$

Once the shapes of the individual basis functions have been defined, the warping function φ is fully determined through the values x^a . The continuous problem of finding a function

ϕ reduces to finding the vectors x^1, \dots, x^n that minimize E . The x^a are thus the degrees of freedom (DOF) of our optimization problem.

Mesh Representation The domain Ω is represented as a mesh consisting of m elements K^1, \dots, K^m , and q nodes at positions X^1, \dots, X^q . Computing the derivative of Eq. (1) with respect to X gives us F as a linear combination of x^a . The energy E can now be formulated as a sum over element integrals:

$$E = \sum_{k=1}^m \int_{K^k} \Psi \left(\sum_{a=1}^n x^a \frac{\partial N^a}{\partial X} \Big|_X \right)^T dX \quad (2)$$

The per-element integrals can be approximated using a numerical quadrature rule. For implementation details, we refer to Hughes [Hug00]. What makes this computation efficient is the fact that basis functions N^a are *local*: inside any element K^k , only a constant number of basis functions can be non-zero, and the sum over a can be reduced to a sum over those basis functions.

If we treat each pixel of an image as an element with a single constant basis function, and introduce some additional terms to handle the resulting discontinuities between elements, we are in fact able to recover the standard finite difference scheme that is used in existing image warping methods.

Numerical Minimization For general $\Psi(F)$, Eq. (2) is a non-linear equation in x^a , and we can minimize it using a Newton method [NW00]. For this, we need the first and second derivatives of E with respect to the x^a , which we compute using either a code generation tool or automatic differentiation (AD). The second derivatives of E result in a sparse, symmetric matrix H . Merging all the vector DOFs x^a into one big vector d of length $2n$ and denoting the first derivative of E with respect to d by f , a single Newton step computes the increment Δd of d by solving the linear system

$$H\Delta d = -f \quad (3)$$

using a direct solver for sparse positive definite systems [SGFS01]. We refer to Appendix A for pseudocode implementation details.

4. Application Specifics

We have now defined image warping in the continuous sense and presented the generic framework of non-linear FEM. At this point we describe choices for the basis functions N^a , the mesh connectivity K^1, \dots, K^m , and the energy density function Ψ based on the application.

4.1. Deformation Energy Densities

In image retargeting, one of the main differences between approaches comes down to what deformation energy function Ψ is used. We look at several different possible choices

for Ψ and show how our continuous formulation allows us to not only reproduce existing warping energies used in earlier work, but also more clearly understand their limitations and design new, improved energy densities.

First we define some useful terms. Most of the commonly used deformation energy densities for image warping can be computed from a combination of quantities derived from the deformation gradient $F(X)$. F is also used, as it tells us how an infinitesimal line segment dX at position X gets deformed under ϕ . The deformed line segment dx can be computed as $dx = FdX$. The Jacobian determinant J is also useful, as it tells us how an infinitesimal area changes under the deformation ϕ . This is defined as:

$$J(X) = \det(F(X)) \quad (4)$$

The right Cauchy-Green tensor C , defined as

$$C(X) = F(X)^T F(X), \quad (5)$$

is invariant under rotation and thus any energy density that can be expressed in terms of C will inherit its rotational invariance. Analyzing the eigenvalues λ_1, λ_2 of $C(X)$, one can see that $\sqrt{\lambda_1}$ and $\sqrt{\lambda_2}$ are the *principal stretches* of the deformation at X , i.e. the minimum and maximum values for the stretch that can be achieved for any direction dX at position X [SSGH01].

Defining and Improving Existing Choices of Ψ

Shamir and Sorkine [SS09] described a simple energy density for image warping that penalizes all deformations except for translations

$$\Psi_F = \|F - I\|^2, \quad (6)$$

where I denotes the 2×2 identity matrix and $\|\cdot\|$ denotes the Frobenius norm of a matrix. Ψ_F is only quadratic in derivatives of ϕ , so when using this energy density, the minimum of a discretized deformation energy E can be found in a single Newton step. For energies that result in more complex optimization problems, existing methods usually employ alternating iterative methods [SS09] that solve a linear least-squares problem while fixing the values of some quantities, like a uniform scale. On the other hand, once we have found the corresponding continuous formulation of such energies, our FEM framework can then solve the resulting non-linear problem in a consistent way, taking all variables and all constraints into account in each Newton step.

Wang et al. [WTSL08] and Laffont et al. [LJW*10] both use a deformation energy that penalizes all transformations other than translation and uniform scaling. The corresponding continuous energy density can be written as:

$$\Psi_{\bar{F}} = \|F - J^{\frac{1}{2}}I\|^2 \quad (7)$$

The right Cauchy-Green tensor C is equal to the identity matrix for pure rotations, a fact we can use to find a rotation-

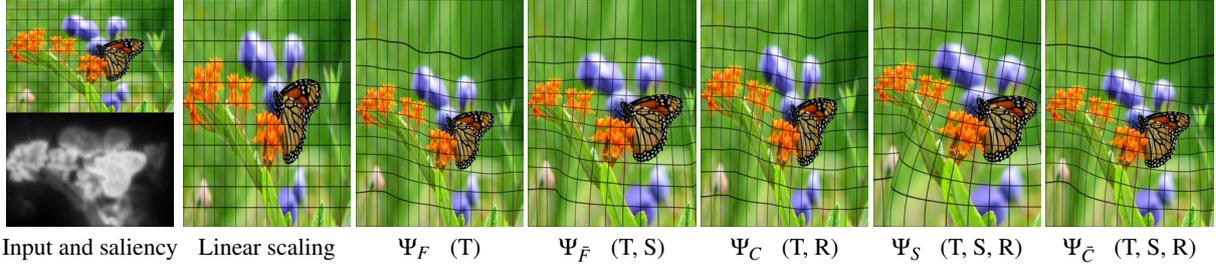


Figure 3: Comparison between different deformation energy density functions used in image retargeting, with indicated invariance to translation (T), scaling (S), and rotation (R). Our novel deformation energy $\Psi_{\bar{C}}$ allows for T,S,R invariance while inherently preventing self-intersections (visible in the lower left half of the Ψ_S image).

invariant energy density [WLSL10]:

$$\Psi_C = \|C - I\|^2 \quad (8)$$

The distortion energy used in Zhang et al. [ZCHM09] also adds to this, scale invariance, permitting elements to undergo a similarity transform (quadratic in derivatives of ϕ). Its corresponding energy density is:

$$\Psi_S = \text{tr}(C) - 2J \quad (9)$$

While this allows for an efficient minimization of E in a single Newton step, this energy density has the drawback of tolerating inversions. This is a significant problem in image retargeting, as it creates visible artifacts after warping (as shown in Figure 3). Existing solutions to this problem usually involve iteratively enforcing expensive non-linear constraints [KLHG09, WTSLO8].

Using our continuous formulation, we can design an improved energy density $\Psi_{\bar{C}}$ that increases the penalty to infinity as the horizontal scaling factor s approaches zero:

$$\Psi_{\bar{C}} = J^{-2} \|C\|^2 \quad (10)$$

This prevents the Newton solver from ever taking a step that would invert an element, and instead results in the line search finding an increment Δd such that J remains positive at every

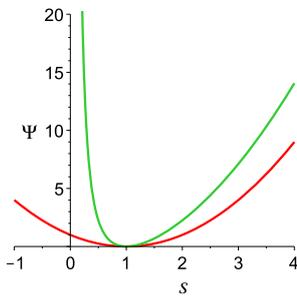


Figure 4: Comparison between the energy densities $\Psi_{\bar{C}} - 2$ (green) and Ψ_S (red) for non-uniform scaling with $F = \begin{pmatrix} s & 0 \\ 0 & 1 \end{pmatrix}$.

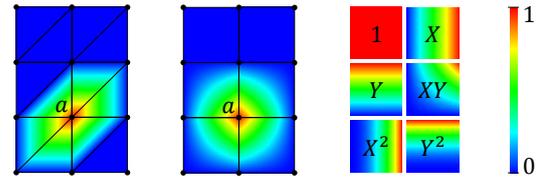


Figure 5: Values of nodal basis function N^a in a triangle mesh (left) and a quad mesh (middle), and the six basis functions used for quadratic DG FEM elements (right).

quadrature point. Inversions are thus prevented by the deformation energy itself, instead of requiring the addition of new constraints. Figure 4 shows a comparison between these two energy densities.

Using $\|C\|^2 = \lambda_1^2 + \lambda_2^2$ and $J^2 = \lambda_1 \lambda_2$, one can derive an alternative representation of $\Psi_{\bar{C}}$,

$$\Psi_{\bar{C}} = \frac{(\lambda_1 - \lambda_2)^2}{\lambda_1 \lambda_2} + 2, \quad (11)$$

showing that this energy density measures the ‘non-uniformity’ of the scaling.

A key component of many content-aware image warping methods is the inclusion of a visual saliency term. The higher the saliency of a region, the better it should be ‘preserved’ in the warp. This can be easily included in our formulation by multiplying the energy density by some spatially varying saliency function $\gamma(X) > 0$. For the results in this paper, we find γ using the method of Goferman et al. [GZMT10].

4.2. Basis Functions

In standard FEM, nodal DOFs are used, meaning that $q = n$ and every DOF is associated with a node: the DOF x^a directly represents the solution of the warp at node a . This implies that for all a , the basis function $N^a(X)$ assumes a value of 1 at position X^a , and a value of 0 at all X^b with $b \neq a$. The natural choice for basis functions for simple triangle and

quad elements are the linear and bi-linear basis functions, respectively (see Figure 5). These basis functions are used in most prior work, and restrict the methods to averaging content (such as image saliency) within elements. We propose the use of discontinuous Galerkin (DG) FEM to allow each element to be endowed with its own set of basis functions. The big advantage of the method is that we are no longer restricted to nodal basis function but we can use simple polynomials of arbitrary order, where the order of polynomials can be chosen independently for each element. Such higher order basis functions are not only able to better represent the solution to the continuous problem, they also take more of the underlying saliency information into account during numerical integration. While higher order basis functions can also be realized in standard FEM, they are almost trivial to implement in DG FEM as they do not depend on the element shapes or mesh connectivity.

For example, the basis functions $1, X_1, X_2, X_1^2, X_1X_2, X_2^2$ allow an element to approximate the solution quadratically (see Figure 5).

To restore the coupling between elements (i.e. introducing “glue”), some additional terms are necessary. To accomplish this using DG FEM, we replace the bi-valued function φ by the so-called *numerical flux* $\hat{\varphi}$ on the edges between elements. This method is based on a DG formulation for non-linear elasticity [TEL06], using the numerical flux of Bassi and Rebay [BR97].

For rendering elements with higher-order basis functions, we triangulate each element using a fine triangulation and compute the new positions of the triangle mesh vertices by evaluating φ inside the element.

By allowing for higher order basis functions, we are able to achieve smoother warps, more efficiently using the high resolution saliency information even with a small number of elements. Figure 6 shows an example where the existing approach results in visible ‘kinks’ between elements, but our higher order basis functions yield smooth transitions between elements.

4.3. Adaptive Meshes

Unlike finite-difference approaches, the mesh-independent continuous formulation provided by our FEM approach allows us to trivially extend our method to arbitrary content-adapted meshes. This allows us to drastically reduce the number of DOFs without noticeable degradation of warping quality, enabling among other things, temporally stable solutions for video examples.

We present two content-aware meshing techniques, one for standard FEM approaches, and the other for DG FEM implementations. In the first, we compute a Delaunay triangulation [She96] of a point set distributed according to *variance* in saliency. This creates increased resolution in areas that

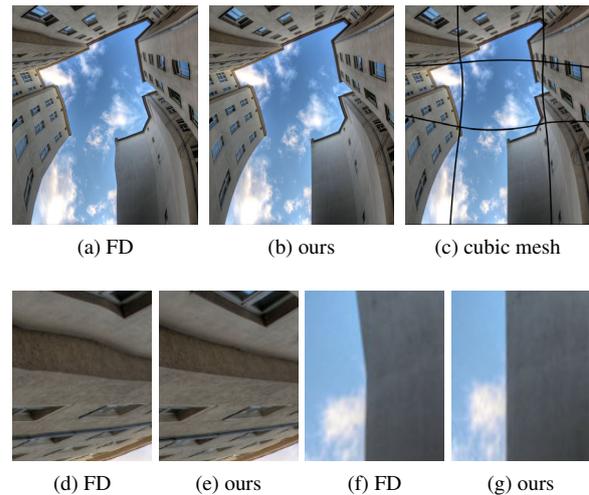


Figure 6: Retargeting using finite differences (FD) with 81 bilinear quads (100 DOFs) vs. our FEM warping using 9 quads with cubic bases (90 DOFs), and the corresponding deformed mesh (c). Our method allows for smoother warping even with fewer DOF.



Figure 7: Error-based adaptivity for DG FEM with linear, quadratic and cubic quad elements (approx. 1024 DOFs each).

are most likely to contain changes in local deformation. An example of this mesh is visible in the teaser.

However, using DG FEM again allows for some additional freedom in mesh construction. In particular, the edges of neighboring elements are not required to coincide, it allows for hanging nodes, and therefore also adaptive quadtree meshes. As the amount of discontinuities between elements is a direct indicator of the local error of the solution, we refine elements with the highest discontinuities, reducing global error in a greedy fashion. This results in a mesh where the function values on edges can safely be averaged for rendering. See Figure 7 for an example.

The advantage of the subdivisions becomes clear in the teaser. Where we get a similar result as the finite-difference approach, but using far fewer degrees of freedom.

We have presented three ways that our approach can improve the quality of existing methods. However, for completeness,



Figure 8: Line constraints controlling the appearance of objects in a modified perspective. Input image with triangle mesh and constrained lines shown in red (left), warp without line constraints (middle), warp with line constraints (right).

there remain some additional commonly used constraints in image retargeting that need to be described, which we discuss next.

4.4. Addressing Additional Constraints

For image retargeting, prior work often constrains all nodes on the boundary of Ω such that $(x_1^a, x_2^a)^T = (X_1^a s_1, X_2^a s_2)^T$ if the image is stretched by $(s_1, s_2)^T$, or allows nodes to slide on the boundary, by constraining $x_1^a = X_1^a s_1$ for nodes with $X_1^a = 0$ or $X_1^a = 1$ and similarly for the X_2 direction. These constraints are implemented as hard constraints by modifying the first and second derivatives of E such that the constrained DOFs do not get modified during a Newton step.

For certain applications like camera stabilization [LGJA09] and stereoscopic disparity editing [LHW*10], weakly enforced point constraints are used (i.e. P_W gets warped to a specific position p_W). Such constraints can be realized by adding an energy term for each point constraint:

$$E_W = \gamma_W \frac{1}{2} \|\phi(P_W) - p_W\|^2 \quad (12)$$

We can also impose line constraints on the warp to weakly enforce straight lines to remain straight after the warp [KLHG09], or to warp initially curved lines to straight ones [CAA09]. Similar to previous approaches, we parameterize the best-fitting straight line as $\sin(\alpha)x_1 + \cos(\alpha)x_2 + b = 0$. For each line constraint, an additional energy term is added to our minimization problem, computed as the integral of the squared distance between the warped curve and the closest point on the fitted straight line, weighted by a penalty γ_L :

$$E_L = \frac{1}{2} \gamma_L \int_0^l (\sin(\alpha)p_1(s) + \cos(\alpha)p_2(s) + b)^2 ds \quad (13)$$

We use per-element quadrature to evaluate this integral. These kind of constraints can be incorporated seamlessly into our method: the two new unknowns α and b parameterizing the fitted straight line simply become two new scalar DOFs of our non-linear problem, and no further special treatment is necessary. The result of these line constraints is shown in Figure 8.



Figure 9: Frames from retargeted video sequences. Showing the input frame, linear scaling, FEM warping, and the corresponding mesh.



(a) original (b) [KLHG09] (c) ours

Figure 10: Comparison of temporal stability in video retargeting. Windowed approaches to stability cannot predict the position of future salient objects. By retargeting entire videos, our method produces more stable output.

5. Results

We show the results from this general framework on two image warping applications. Please refer to the supplementary video for full examples of these methods.

Video Retargeting We use our proposed inversion preventing energy density function Ψ_C for retargeting examples. To solve over a whole sequence of frames and still keep the problem at a tractable size, we have used an adaptive triangle mesh with the standard FEM method. See Figure 9 for examples.

We compare the temporal stability of our method to prior work by Krähenbühl et al. [KLHG09] in Figure 10, which is restricted to enforcing only local smoothness between frames. Please refer to the supplemental video to see the full sequence, where differences are more apparent.

Video Stabilization Warping-based video stabilization methods [LGJA09] consist of tracking feature points over time, reconstructing their positions and the camera in 3D space, then reprojecting feature points into a new, stabilized camera path. The input to the image warp consists of a set of weighted feature points with source and (reprojected) target positions, which guide the warp as weak point constraints. There is no direct influence between the warps computed for individual frames, so each frame can be warped independently and a regular mesh of size 64×36 as used in Liu et al. [LGJA09] provides a good quality-vs-performance trade-off. See Figure 11 for examples.

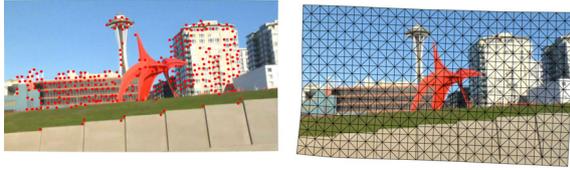


Figure 11: Our image warping method applied to camera stabilization. Input frame with detected features (left), stabilized frame with FEM mesh (right).

5.1. Timing

We show the timing of our method on several examples presented in this paper in Table 1. These results were generated on an Intel Core i7 3.2 GHz computer, in a single-threaded application.

Example	#Els	#Frames	#Newton	t_{pre}	t_{Hf}	t_{solve}
Fig. 1, right	2612	1	7	60	2	19
Fig. 9, top	111124	234	1	27820	600	6311
Fig. 9, bottom	54249	117	1	14634	273	2537
Fig. 11	1152	1	2	31	2	4

Table 1: Problem complexity and timings (in ms) for pre-computation (including mesh generation), computation of H , f , and solving the linear system.

6. Discussion

In conclusion, we have presented a novel, general representation for image warping that unifies a wide range of existing solutions. Our approach provides a well defined continuous mathematical formulation that has multiple *real-world* advantages. For one, a mathematical basis allows for energy densities to be clearly defined and analyzed, allowing for improved understanding and design. Our representation is independent of mesh formulation, which allows for simple extensions for adaptive meshing and temporally stable solutions. Additionally, a method based on DG FEM is presented which simplifies adaptive meshing and the use of higher order basis functions for smoother warps. It comes at the cost of a more complex implementation and the need to average displacements at edges in order to avoid discontinuous warps. However, probably most significantly, there is a wide range of literature and ongoing research about FEM techniques in the mechanics and geometry communities, and by phrasing the image warping problem in the same context, both areas of research have the potential to benefit tremendously from their combined research efforts.

One of the main limitations of our FEM approach is that it can be more complex to implement than traditional finite-difference methods. However, this is a one-time cost, and when completed, the framework is very flexible, making application to novel problems and domains a much simpler

task than before. Furthermore, many FEM implementations exist for mesh deformation, and could be easily extended to image-warping problems.

In this paper, we have only scratched the surface of what the FEM could be used for in the context of image editing applications. Further insight into FEM and related methods could provide additional capabilities, such as using X-FEM for discontinuous warping methods. Our representation is also not restricted to 2D elements, and one extension could be a mesh subdivision with 3D elements, such as a video cube oct-tree, that may provide stability for temporal solutions. In addition, it is possible that energies with higher order derivatives could be useful, e.g. for the spherical distortion application.

Acknowledgments

We would like to thank Manuel Lang for helpful discussions and Feng Liu for providing a comparison to his stabilization method. Copyrights of the source videos belong to Mammoth HD, Inc and ARS Film Production. Images were taken from the RetargetMe dataset [RGSS10]. All copyrights belong to the respective owners.

References

- [BN92] BEIER T., NEELY S.: Feature-based image metamorphosis. In *SIGGRAPH* (1992), pp. 35–42. 2
- [BR97] BASSI F., REBAY S.: A high-order accurate discontinuous finite element method for the numerical solution of the compressible navier-stokes equations. *J. Comput. Phys.* 131 (1997), 267–279. 6
- [BW97] BONET J., WOOD R. D.: *Nonlinear Continuum Mechanics for Finite Element Analysis*. Cambridge University Press, 1997. 2
- [CAA09] CARROLL R., AGRAWALA M., AGARWALA A.: Optimizing content-preserving projections for wide-angle images. *ACM Trans. Graph.* 28, 3 (2009). 2, 7
- [CAA10] CARROLL R., AGARWALA A., AGRAWALA M.: Image warps for artistic perspective manipulation. *ACM Trans. Graph.* 29, 4 (2010). 2
- [Coc03] COCKBURN B.: Discontinuous Galerkin methods. *Z. Angew. Math. Mech.* 80, 11 (2003), 731–754. 2
- [Gee94] GEE J. C.: Finite element approach to warping of brain images. *Proceedings of SPIE* (1994), 327–337. 2
- [GWCO09] GUTTMANN M., WOLF L., COHEN-OR D.: Semi-automatic stereo extraction from video footage. In *Computer Vision, 2009 IEEE 12th International Conference on* (Oct 2009), pp. 136–142. 3
- [GZMT10] GOFERMAN S., ZELNIK-MANOR L., TAL A.: Context-aware saliency detection. In *CVPR* (2010), pp. 2376–2383. 5
- [Hug00] HUGHES T. J. R.: *The Finite Element Method. Linear Static and Dynamic Finite Element Analysis*. Dover Publications, 2000. 4
- [JTSZ10] JACOBSON A., TOSUN E., SORKINE O., ZORIN D.: Mixed finite elements for variational surface modeling. *Comput. Graph. Forum* 29, 5 (2010), 1467–8659. 2

- [KLHG09] KRÄHENBÜHL P., LANG M., HORNUNG A., GROSS M. H.: A system for retargeting of streaming video. *ACM Trans. Graph.* 28, 5 (2009). 3, 5, 7
- [KMBG08] KAUFMANN P., MARTIN S., BOTSCH M., GROSS M.: Flexible simulation of deformable models using discontinuous galerkin fem. In *Proc. of Symp. on Computer Animation* (2008), pp. 105–115. 2
- [LGJA09] LIU F., GLEICHER M., JIN H., AGARWALA A.: Content-preserving warps for 3d video stabilization. *ACM Trans. Graph.* 28, 3 (2009). 2, 7
- [LHW*10] LANG M., HORNUNG A., WANG O., POULAKOS S., SMOLIC A., GROSS M. H.: Nonlinear disparity mapping for stereoscopic 3d. *ACM Trans. Graph.* 29, 4 (2010). 2, 7
- [LJW*10] LAFFONT P.-Y., JUN J. Y., WOLF C., TAI Y.-W., IDRISSE K., DRETTAKIS G., EUI YOON S.: Interactive content-aware zooming. In *Proceedings of the Graphics Interface 2010 Conference* (2010), Mould D., Noel S., (Eds.), pp. 79–87. 3, 4
- [NLLG10] NIU Y., LIU F., LI X., GLEICHER M.: Warp propagation for video resizing. In *CVPR* (2010), pp. 537–544. 3
- [NW00] NOCEDAL J., WRIGHT S. J.: *Numerical Optimization*. Springer, August 2000. 4
- [PWS12] PANOZZO D., WEBER O., SORKINE O.: Robust image retargeting via axis-aligned deformation. *Computer Graphics Forum (proceedings of EUROGRAPHICS)* 31, 2 (2012), 229–236. 3
- [RGSS10] RUBINSTEIN M., GUTIERREZ D., SORKINE O., SHAMIR A.: A comparative study of image retargeting. In *ACM SIGGRAPH Asia 2010 papers* (New York, NY, USA, 2010), SIGGRAPH ASIA '10, ACM, pp. 160:1–160:10. 8
- [SGFS01] SCHENK O., GÄRTNER K., FICHTNER W., STRICKER A.: PARDISO: a high-performance serial and parallel sparse linear solver in semiconductor device simulation. *Future Generation Computer Systems* 18, 1 (2001), 69–78. 4
- [She96] SHEWCHUK J. R.: Triangle: Engineering a 2D Quality Mesh Generator and Delaunay Triangulator. In *Applied Computational Geometry: Towards Geometric Engineering*, vol. 1148. Springer-Verlag, May 1996, pp. 203–222. 6
- [SLS*07] SHARF A., LEWINER T., SHKLARSKI G., TOLEDO S., COHEN-OR D.: Interactive topology-aware surface reconstruction. *ACM Trans. Graph.* 26, 3 (2007), 43. 2
- [SS09] SHAMIR A., SORKINE O.: Visual media retargeting. In *SIGGRAPH ASIA Courses* (2009). 2, 4
- [SSGH01] SANDER P. V., SNYDER J., GORTLER S. J., HOPPE H.: Texture mapping progressive meshes. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques* (2001), SIGGRAPH '01, ACM, pp. 409–416. 4
- [TEL06] TEN EYCK A., LEW A.: Discontinuous galerkin methods for non-linear elasticity. *Int. J. Numer. Meth. Engng.* 67, 9 (2006), 1204–1243. 6
- [TF88] TERZOPOULOS D., FLEISCHER K.: Deformable models. *The Visual Computer* 4 (1988), 306–331. 2
- [WFS*09] WANG Y.-S., FU H., SORKINE O., LEE T.-Y., SEIDEL H.-P.: Motion-aware temporal coherence for video resizing. *ACM Trans. Graph.* 28, 5 (2009). 3
- [WLSL10] WANG Y.-S., LIN H.-C., SORKINE O., LEE T.-Y.: Motion-based video retargeting with optimized crop-and-warp. *ACM Trans. Graph.* 29, 4 (2010). 3, 5
- [WTSL08] WANG Y.-S., TAI C.-L., SORKINE O., LEE T.-Y.: Optimized scale-and-stretch for image resizing. In *ACM SIGGRAPH Asia 2008 papers* (2008), SIGGRAPH Asia '08, ACM, pp. 118:1–118:8. 4, 5

- [ZCHM09] ZHANG G.-X., CHENG M.-M., HU S.-M., MARTIN R. R.: A shape-preserving approach to image resizing. *Computer Graphics Forum* 28, 7 (2009), 1897–1906. 5

Appendix A: Implementation Notes

The function *ComputeE* shown below computes the energy E for given DOF values d . It calls *ComputePsi* to evaluate the energy density Ψ for the given deformation gradient F . *ComputeBFunDeriv* computes the derivative of a basis function at a given position. E_C computes the energy of constraint C for the given DOF values.

```

1 E = ComputeE( d = (x1T, ..., xnT)T )
2   E ← 0
3   for each element Kk:
4     for each quad. point qi with weight wi of Kk:
5       F ← 0
6       for each basis function Na of Kk:
7         b ← ComputeBFunDeriv(a, qi)
8         F += xa bT
9       end
10      E += ComputePsi(F) wi
11    end
12  end
13  for each soft constraint C:
14    E += EC(d)
15  end
16 end

```

Given initial values for the DOFs d_{init} that satisfy the hard constraints, the function *Solve* shown below finds a local minimum of the energy E . *LinSolve* solves a system of linear equations. *ApplyHardConstraints* modifies H and f such that for all constrained DOFs i , $f_i = 0$, $H_{ii} = I$ and $H_{ij} = H_{ji} = 0$ for $j \neq i$. *ComputeHf* computes the first and second derivatives of E with respect to the DOFs, which similarly to *ComputeE* can be evaluated on a per-element basis.

```

1 E = Solve(dinit)
2   d ← dinit
3   do
4     E ← ComputeE(d)
5     H, f ← ComputeHf(d)
6     H, f ← ApplyHardConstraints(H, f)
7     Find small β ≥ 0 s.t. H + βI is pos. def.
8     Δd ← LinSolve((H + βI)Δd = -f)
9     Find large α ≤ 1 s.t. ComputeE(d + αΔd) < E
10    d += αΔd
11  while ||f||2 > tol
12 end

```