Fast Dynamic Programming for Elastic Registration of Curves

Javier Bernal, Gunay Dogan, Robert C. Hagwood, NIST, Theiss Research

National Institute of Standards and Technology
U.S. Department of Commerce

IEEE 2016 Conference on Computer Vision and Pattern Recognition
CVPR 2016

Goal: Fast registration of curves using dynamic programming (DP) for the efficient computation of elastic shape distances, alignment of functions, etc.

Our contribution: Development and implementation of linear DP algorithm for computing optimal diffeomorphisms for elastic registration of curves.

Elastic registration formulation:

For $F:[0,1]\times[0,1]\times\mathbb{R}\to\mathbb{R}$, minimize energy

$$E(\gamma)=\int_0^1 F(t,\gamma(t),\dot\gamma(t))dt$$

with respect to γ, γ a diffeomorphism of $[0,1]$ onto itself $(\gamma(0)=0,\gamma(1)=1,\dot\gamma>0)$.

A typical function F (from elastic shape analysis):

$$F(t,\gamma(t),\dot\gamma(t))=\|R(\theta)q_1(t+t_0)-\sqrt{\dot\gamma(t)}q_2(\gamma(t))\|^2.$$

$t_0\equiv$ seed or starting point.

$R(\theta)\equiv 2\times 2$ rotation matrix defined by angle θ.

$q_i(t)\equiv\dot\beta_i(t)/\|\dot\beta_i(t)\|^{1/2},\ i=1,2.$

$\beta_i:[0,1]\to\mathbb{R}^2$, C^2 closed curve of unit length.

Discretized formulation:

$$E(\vec\gamma)=\frac{1}{2}\sum_{l=1}^{N-1}(t_{l+1}-t_l)(F(t_{l+1},\gamma_{l+1},\dot\gamma_{l+1})+F(t_l,\gamma_l,\dot\gamma_l)).$$

$\{t_l\}_{l=1}^N\equiv$ a partition of $[0,1]$.

$\vec\gamma\equiv\{\gamma_l\}_{l=1}^N\equiv\{\gamma(t_l)\}_{l=1}^N,\ \{\dot\gamma_l\}_{l=1}^N\equiv\{\dot\gamma(t_l)\}_{l=1}^N.$

Approximating γ **with a grid:**

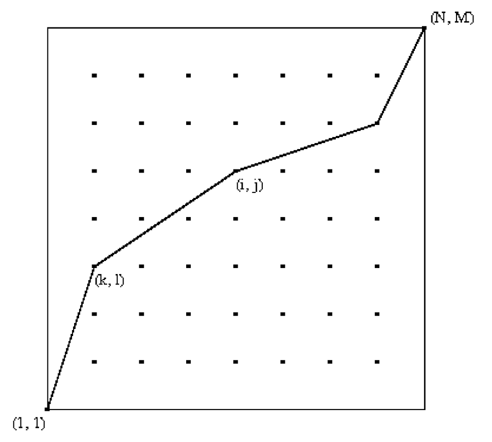
$\{t_i\}_{i=1}^N\equiv$ a partition of $[0,1]$ (x axis).

$\{z_j\}_{j=1}^M\equiv$ a partition of $[0,1]$ (y axis).

Point (t_i,z_j) coincides with grid point (i,j).

γ approximated by increasing piecewise linear γ^*.

Vertices of graph of γ^* are grid points.



Energy over line segment $\overline{(k,l)(i,j)}$:

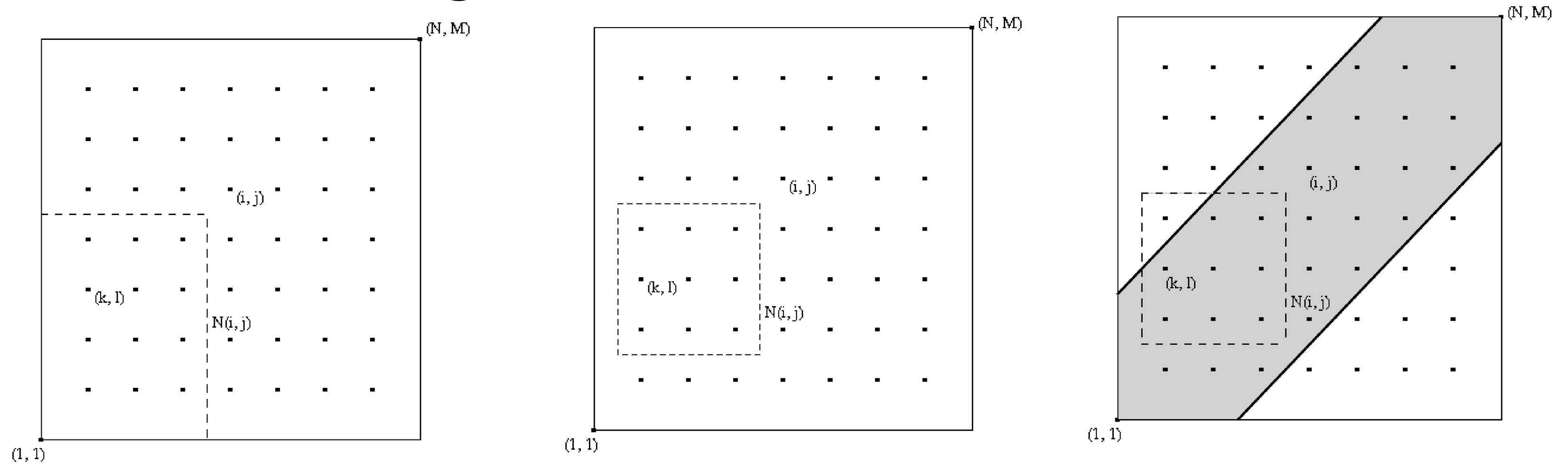
$$E_{(k,l)}^{(i,j)}=\frac{1}{2}\sum_{m=k}^{i-1}(t_{m+1}-t_m)(F_{m+1}+F_m).$$

$F_m\equiv F(t_m,\alpha(t_m),L),\ m=k,\dots,i.$

$\alpha\equiv$ linear function with $\alpha(t_k)=z_l,\ \alpha(t_i)=z_j.$

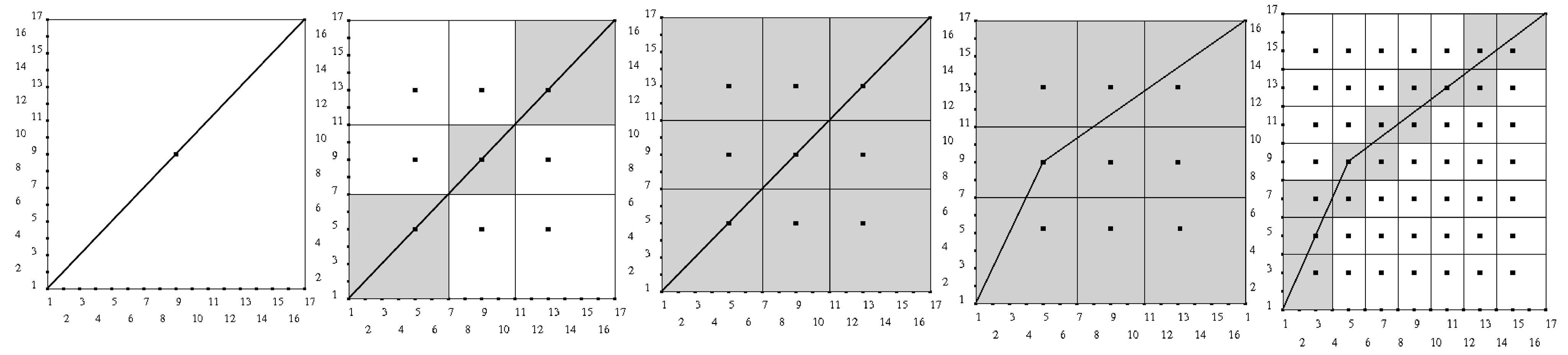
$L\equiv$ slope of line segment $\overline{(k,l)(i,j)}$.

Known DP algorithms:

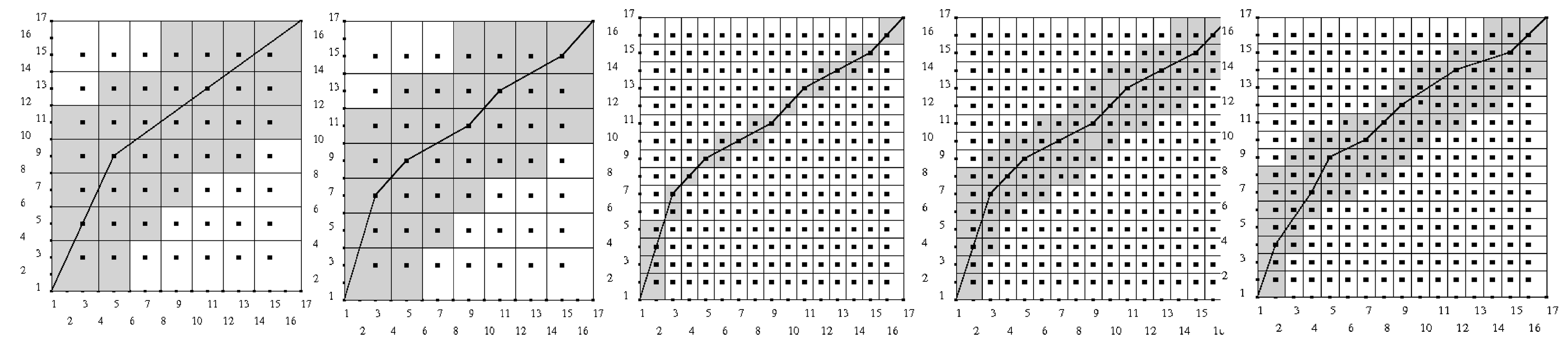


Left: the $O(N^4)$ DP algorithm guaranteed to compute optimal solution.
Center: the $O(N^2)$ original DP algorithm. Right: the $O(N^2)$ fast DP algorithm.
In all cases, $N(i,j)$ is a trailing set of (i,j) and the optimal energy at (i,j) relative to $N(i,j)$ is $E(i,j)=\min_{(k,l)\in N(i,j)}(E(k,l)+E_{(k,l)}^{(i,j)})$.

New DP algorithm: O(N) adapt DP algorithm using strips of width O(1):



From 3 x 3 subgrid to 5 x 5 subgrid to 9 x 9 subgrid



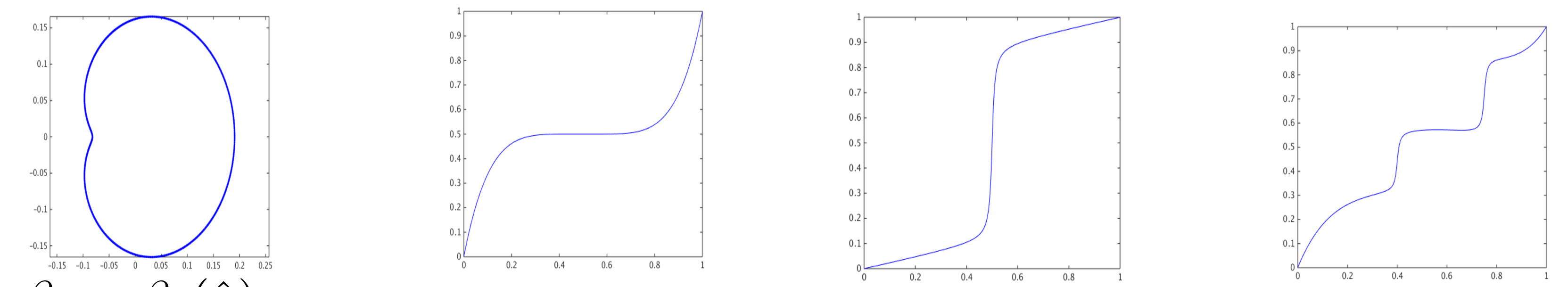
to 17 x 17 subgrid

Complexity = O(N + N/2 + N/4 …) = O(N)

Numerical results:

Diffeomorphism recovery:

$\beta_2\equiv$ a limacon | Flat $\hat\gamma$ | Steep $\hat\gamma$ | Bumpy $\hat\gamma$



$\beta_1\equiv\beta_2(\hat\gamma).$

$$F(t,\gamma(t),\dot\gamma(t))=\|q_1(t)-\sqrt{\dot\gamma(t)}q_2(\gamma(t))\|^2,\quad q_i(t)=\dot\beta_i(t)/\|\dot\beta_i(t)\|^{1/2},\ i=1,2.$$

	$N=$	64	128	256	512	1024	2048
	$\omega=$	64	32	16	12	12	12
$original-DP$	all $\hat\gamma$	0.49	1.26	1.00	2.07	8.48	34.3
$fast-DP$	all $\hat\gamma$	0.24	0.66	0.51	1.04	4.20	17.0
$adapt-dP$	Flat $\hat\gamma$	0.65	0.67	0.27	0.29	0.57	1.20
	Steep $\hat\gamma$	0.65	0.56	0.28	0.31	0.62	1.32
	Bumpy $\hat\gamma$	0.81	1.04	0.37	0.36	0.70	1.46

Times in seconds. $\omega\equiv$ number of grid points per side of $N(i,j)$.
In all cases, L^2 error between true solution $\hat\gamma$ and computed γ^* $\left(\frac{1}{N-1}\left(\sum_{l=1}^{N-1}(\hat\gamma(t_l)-\gamma^*(t_l))^2\right)^{\frac{1}{2}}\right)$ was less than 10^{-3}.

Function alignment by warping:

$$F(t,\gamma(t),\dot\gamma(t))=(q_1(t)-\sqrt{\dot\gamma(t)}q_2(\gamma(t)))^2.$$

$q_i(t)\equiv\text{sign}(\dot f_i(t))\sqrt{|\dot f_i(t)|}$, absolutely continuous $f_i:[0,1]\to\mathbb{R},\quad i=1,2.$

Function 1 in blue has 19,693 points, function 2 in red has 19,763 points.

Function 1 was aligned to function 2 in **172 seconds.**

Warping function γ on the right inside last strip (boundary in green).