Supervised Descent Method and its Applications to Face Alignment

Xuehan Xiong, Fernando De la Torre (CVPR-2013, extensions: submitted to TPAMI)

Zoltán Szabó

Gatsby Unit, Tea Talk

March 16, 2015

Motivation

- Computer vision: many tasks boil down to continuous nonlinear optimization.
- Our focus: facial feature detection/tracking.

Newton's method (and its variants)

- Task: $\min_{\mathbf{x}} f(\mathbf{x})$, $f \in C^2$.
- Newton's method: locally quadratic approximation,
  - $\mathbf{x}_0$: given.
  - Second order Taylor expansion ($k = 0, 1, 2, ...$) around $\mathbf{x}_k$:

$$f(\mathbf{x}_k + \Delta\mathbf{x}) \approx f(\mathbf{x}_k) + \mathbf{J}_f(\mathbf{x}_k)^T(\Delta\mathbf{x}) + \frac{1}{2}(\Delta\mathbf{x})^T\mathbf{H}(\mathbf{x}_k)(\Delta\mathbf{x}) \Rightarrow$$

$$\Delta\mathbf{x}_{k+1} = -\mathbf{H}^{-1}(\mathbf{x}_k)\mathbf{J}_f(\mathbf{x}_k),$$

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \Delta\mathbf{x}_{k+1}.$$

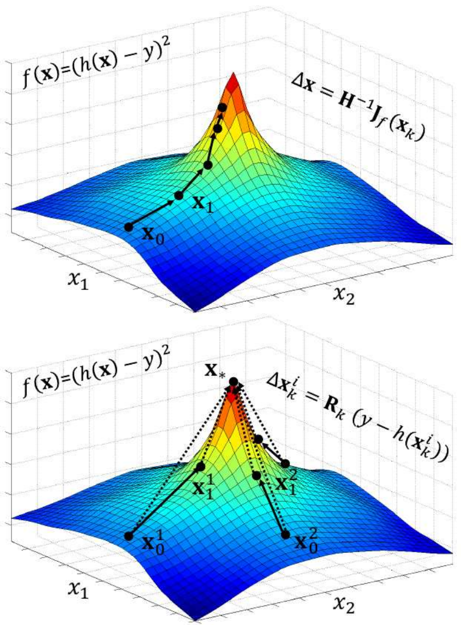
Newton's method: pro & contra

- Advantages:
  - If it converges $\Rightarrow$ quadratic rate ($q = 2$)

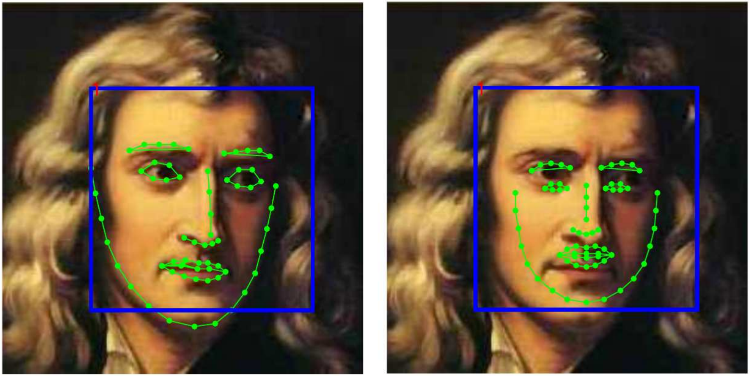
$$\lim_{k\to\infty} \frac{\|\mathbf{x}_k - \mathbf{x}_*\|}{\|\mathbf{x}_{k-1} - \mathbf{x}_*\|^q} = L > 0.$$

  - If $\mathbf{x}_0$ is "close enough" to $\mathbf{x}_*$ $\Rightarrow$ convergence.
- Disadvantages (in CV):
  - f: non-differentiable (SIFT) $\rightarrow$ numerical $\mathbf{J}_f, \mathbf{H}$: slow.
  - Linear equation: expensive (quasi too).

Optimization idea (z-axis: reversed)



Face alignment: $\mathbf{x}_0$ – initial estimation, $\mathbf{x}_*$ – manual labels



(a) $\mathbf{x}_*$ (b) $\mathbf{x}_0$

Face alignment: formulation

- Image: $\mathbf{d}$.
- Landmark locations (p): $\mathbf{x} = [x_1; y_1; \ldots; x_p; y_p] \in \mathbb{R}^{2p}$.
- Feature extraction function (SIFT): $\mathbf{h}$.
- Features around the landmarks: $\mathbf{h}(\mathbf{x}; \mathbf{d}) \in \mathbb{R}^{128p}$.
- Task: for given $\mathbf{x}_0$

$$g(\Delta\mathbf{x}) = f(\mathbf{x}_0 + \Delta\mathbf{x}) = \|\mathbf{h}(\mathbf{x}_0 + \Delta\mathbf{x}; \mathbf{d}) - \phi_*\|_2^2 \to \min_{\Delta\mathbf{x}},$$

$$\phi_* = \mathbf{h}(\mathbf{x}_*; \mathbf{d}).$$

Face alignment

- Task:

$$f(\mathbf{x}_0 + \Delta\mathbf{x}) = \|\mathbf{h}(\mathbf{x}_0 + \Delta\mathbf{x}; \mathbf{d}) - \phi_*\|_2^2 \to \min_{\Delta\mathbf{x}}.$$

- By the Newton trick (+chain rule):

$$\begin{aligned}
\Delta\mathbf{x}_1 &= -\mathbf{H}^{-1}(\mathbf{x}_0)\mathbf{J}_f(\mathbf{x}_0) = -2\mathbf{H}^{-1}(\mathbf{x}_0)\mathbf{J}_{\mathbf{h}}^T(\mathbf{x}_0)(\phi_0 - \phi_*),\\
\phi_0 &= \mathbf{h}(\mathbf{x}_0; \mathbf{d}),\\
\phi_* &= \mathbf{h}(\mathbf{x}_*; \mathbf{d}).
\end{aligned}$$

Face alignment

- Task:

$$f(\mathbf{x}_0 + \Delta\mathbf{x}) = \|\mathbf{h}(\mathbf{x}_0 + \Delta\mathbf{x}; \mathbf{d}) - \phi_*\|_2^2 \to \min_{\Delta\mathbf{x}}.$$

- By the Newton trick (+chain rule):

$$\Delta\mathbf{x}_1 = -\mathbf{H}^{-1}(\mathbf{x}_0)\mathbf{J}_f(\mathbf{x}_0) = -2\mathbf{H}^{-1}(\mathbf{x}_0)\mathbf{J}_{\mathbf{h}}^T(\mathbf{x}_0)(\phi_0 - \phi_*),$$
$$\phi_0 = \mathbf{h}(\mathbf{x}_0; \mathbf{d}),$$
$$\phi_* = \mathbf{h}(\mathbf{x}_*; \mathbf{d}).$$

- Idea [$\mathbf{H} := \mathbf{H}(\mathbf{x}_0)$, $J_h := J_h(\mathbf{x}_0)$]:

$$\Delta\mathbf{x}_1 = -2\mathbf{H}^{-1}\mathbf{J}_{\mathbf{h}}^T(\phi_0 - \phi_*) = \left[-2\mathbf{H}^{-1}\mathbf{J}_{\mathbf{h}}^T\right]\phi_0 + \left[2\mathbf{H}^{-1}\mathbf{J}_{\mathbf{h}}^T\phi_*\right]$$
$$= \mathbf{R}_0\phi_0 + \mathbf{b}_0 \Rightarrow \text{optimize for } (\mathbf{R}_0, \mathbf{b}_0) \text{ based on samples.}$$

- The algorithm is unlikely to converge in 1 iteration.
- Cascade of regressors:

$$\Delta \mathbf{x}_1 = \mathbf{R}_0 \phi_0 + \mathbf{b}_0,$$

$$-- - -$$

$$\Delta \mathbf{x}_k = \mathbf{R}_{k-1} \phi_{k-1} + \mathbf{b}_{k-1}, \text{ where}$$

$$\phi_{k-1} = \mathbf{h}(\mathbf{x}_{k-1}; \mathbf{d}) : \text{features at the previous landmarks.}$$

Face alignment: optimization ($k = 0$)

- Given:
  - set of images: $\{\mathbf{d}^i\}_{i=1}^N$,
  - hand-labelled landmarks: $\{\mathbf{x}_*^i\}_{i=1}^N$,
  - initial estimates: $\{\mathbf{x}_0^i\}_{i=1}^N \Rightarrow$
- Optimal updates, extracted features:

$$\Delta\mathbf{x}_{*0}^i = \mathbf{x}_*^i - \mathbf{x}_0^i, \quad \phi_0^i = \mathbf{h}\left(\mathbf{x}_0^i; \mathbf{d}^i\right).$$

- Objective:

$$J(\mathbf{R}_0, \mathbf{b}_0) = \frac{1}{N}\sum_{i=1}^N \left\|\Delta\mathbf{x}_{*0}^i - \mathbf{R}_0\phi_0^i - \mathbf{b}_0\right\|_2^2 \to \min_{\mathbf{R}_0, \mathbf{b}_0}.$$

Face alignment: optimization (general k)

- Update the landmark estimates ($\mathbf{x}_k$):

$$\Delta \mathbf{x}_k^i = \mathbf{R}_{k-1}\phi_{k-1}^i + \mathbf{b}_{k-1} \quad (i = 1, \ldots, N).$$

- Compute optimal updates ($\forall i$), extract features:

$$\Delta \mathbf{x}_{*k}^i = \mathbf{x}_*^i - \mathbf{x}_k^i, \quad \phi_k^i = \mathbf{h}\left(\mathbf{x}_k^i; \mathbf{d}^i\right).$$

- Objective:

$$J(\mathbf{R}_k, \mathbf{b}_k) = \frac{1}{N}\sum_{i=1}^{N}\left\|\Delta \mathbf{x}_{*k}^i - \mathbf{R}_k\phi_k^i - \mathbf{b}_k\right\|_2^2 \rightarrow \min_{\mathbf{R}_k, \mathbf{b}_k}.$$

Numerical experience: convergence in $4-5$ steps.

Face alignment: training, testing

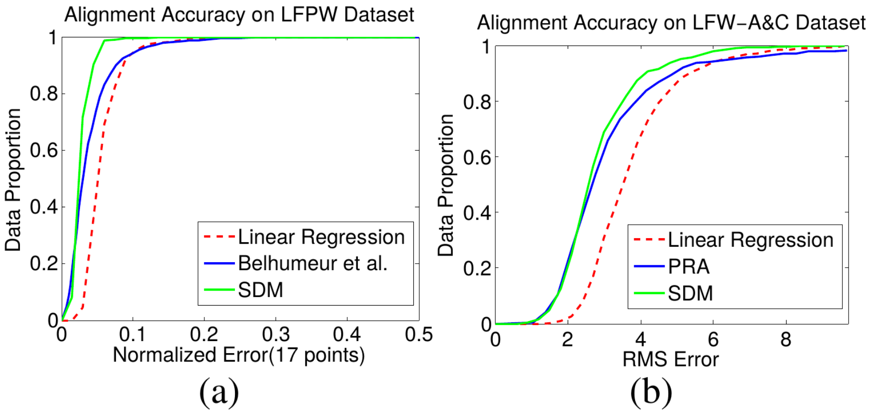
- Training $\Rightarrow \{\mathbf{R}_k, \mathbf{b}_k\}$.
- Testing:
  - test image: $\tilde{\mathbf{d}}$,
  - inital estimate: $\mathbf{x}_0$,
  - extract features: $\phi_0 = \mathbf{h}\left(\mathbf{x}_0; \tilde{\mathbf{d}}\right)$,
  - iteratively compute $\Delta\mathbf{x}_k$, the features at $\mathbf{x}_k$ ($k = 1, \ldots$):

$$\Delta\mathbf{x}_k = \mathbf{R}_{k-1}\phi_{k-1} + \mathbf{b}_{k-1},$$
$$\phi_k = \mathbf{h}\left(\mathbf{x}_k; \tilde{\mathbf{d}}\right).$$

Facial feature detection: 2 ”face in the wild” datasets

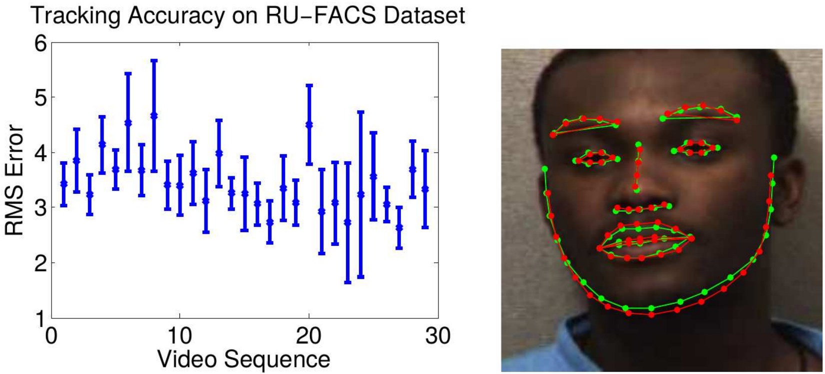
Last row: 10 worst cases.

Facial feature detection: cumulative error distribution curves



Facial feature tracking

- Initialization = landmark estimate from the previous frame.
- (a): average RMSE-s on 29 videos, (b): RMSE=5.03 demo.



(a) (b)

Summary

- Focus: continuous nonlinear optimization.
- Newton's method: expensive.
- Idea:
  - supervised Newton method,
  - learn cascade of affine regressors based on samples.
- Application:
  - facial feature detecion,
  - face tracking.

Thank you for the attention!