Supplementary Material for Paper Submission 307

Anonymous Author(s)

October 12, 2013

Contents

1 Image Formation Model 2
   1.1 Parameterization of Deformation Field $W(x;p)$ ......................... 2
   1.2 The bases function $B(x)$ .............................................. 2
   1.3 Image Patch .............................................................. 2

2 Main Theorem 3
   2.1 Pull-back conditions ..................................................... 3
   2.2 Relaxed Lipchitz Conditions .......................................... 4
   2.3 Guaranteed Nearest Neighbor ......................................... 4
   2.4 Number of Samples Needed ........................................... 6

3 Sampling within a Hypercube 7
   3.1 Covering the Entire Hypercube ....................................... 8
   3.2 Covering a Subspace within Hypercube ............................... 8

4 Finding optimal curve $\gamma = \gamma(\alpha)$ 10

5 More Experiments 11
1 Image Formation Model

We repeat our model here from the paper for readability. The template image $T$ and the distorted image $I_p$ is related by the following equality:

$$I_p(W(x;p)) = T(x)$$

where, $W(x;p)$ is the deformation field that maps the 2D location of pixel $x$ on the template with the 2D location of pixel $W(x;p)$ on the distorted image $I_p$.

1.1 Parameterization of Deformation Field $W(x;p)$

$W(x;p)$ is parameterized by the displacements of $K$ landmarks. Each landmark $i$ has a rest location $l_i$ and displacement $p(i)$. Both of them are 2-dimensional column vectors. For any $x$, its deformation $W(x;p)$ is a weighted combination of the displacements of $K$ landmarks:

$$W(x;p) = x + \sum_{i=1}^{K} b_i(x)p(i)$$

whose $b_i(x)$ is the weight from landmark $i$ to location $x$. Naturally we have $\sum_i b_i(x) = 1$ (all weights at any location sums to 1), $b_i(l_i) = 1$ and $b_i(l_j) = 0$ for $j \neq i$. We can also write Eqn. 2 as the following matrix form:

$$W(x,p) = x + B(x)p$$

where $B(x) = [b_1(x), b_2(x), \ldots, b_K(x)]$ is a K-dimensional column vector and $p = [p(1), p(2), \ldots, p(K)]^\top$ is a K-by-2 matrix. Each row of $p$ is the displacement $p(i)^\top$ of landmark $i$.

1.2 The bases function $B(x)$

Given any pixel location $x$, the weighting function $b_i(x)$ satisfies $0 \leq b_i(x) \leq 1$ and $\sum_i b_i(x) = 1$. For landmark $i$, $b_i(l_i) = 1$ and $b_i(l_j) = 0$ for $j \neq i$.

We assume that $B(x) = [b_1(x), b_2(x), \ldots, b_K(x)]$ is smoothly changing:

**Assumption 1** There exists $c_B$ so that:

$$\|B(x) - B(y)\|_\infty \leq c_B \|x - y\|_\infty$$

Intuitively, Eqn. 3 measures how smooth the bases change over space.

**Lemma 1 (Unity bound)** For any $x$ and any $p$, we have $\|B(x)p\|_\infty \leq \|p\|_\infty$.

**Proof**

$$\|B(x)p\|_\infty = \max\{\sum_i b_i(x)p_i, \sum_i b_i(x)p_i^\top\} \leq \max\{\max_i p_i \sum_i b_i(x), \max_i p_i^\top \sum_i b_i(x)\} = \|p\|_\infty$$

using the fact that $\sum_i b_i(x) = 1$ for any $x$.

1.3 Image Patch

We consider a square $R = R(x,r) = \{y : \|x - y\|_\infty \leq r\}$ centered at $x$ with side $2r$. Given an image $I$ treated as a long vector of pixels, the image content (patch) $I(R)$ is a vector obtained by selecting the components of $I$ that is spatially contained in the square $R$. $S = S(x,r)$ is the subset of landmarks that most influences the image content at $I(R)$. The parameters on the subset are denoted as $p(S)$. Fig. 1 shows the relationship.

Since $p(S)$ is a $|S|$-by-2 matrix, there are at most $2|S|$ apparent degrees of freedom for patch $I(R)$. How large is $|S|$? If landmarks are distributed uniformly (e.g., on a regular grid), $|S|$ is proportional to $Area(R)$, or to the square of the patch scale ($r^2$), which gives $2|S| \propto r^2$.

On the other hand, if the overall effective degree of freedom is $d$, then no matter how large $2|S|$ is, $p(S)$ contains dependent displacements and the effective degree of freedom in $R$ never
Now we need to check the pixel distance between $u$ and $v$. For any Eqn. 4.

$\textbf{Proof}$

For any $y \in R = R(x, r)$, by definitions of Eqn. 6 and Eqn. 7 we have:

$$H(I_p, q)(y) = I_p(W(y; q))$$
$$I_{p-q}(y) = T(W^{-1}(y; p - q)) = I_p(W(W^{-1}(y; p - q), p))$$

Now we need to check the pixel distance between $u = W(y; q)$ and $v = W(W^{-1}(y; p - q), p)$. Note both are pixel locations on distorted image $I_p$. If we can bound $\|u - v\|_\infty$, then from $I_p$’s appearance, we can obtain the bound for $|H(I_p, q)(y) - I_{p-q}(y)|$.

Denote $z = W^{-1}(y; p - q)$ which is a pixel location on the template. By definition we have:

$$y = z + B(z)(p - q)$$
then we have \( \|y - z\|_\infty = \|B(z)(p - q)\|_\infty \leq \|p - q\|_\infty \leq r \) by Lemma I. On the other hand, we have:

\[
\begin{align*}
 p - v &= W(y, q) - W(z, p) \\
 &= y + B(y)q - z - B(z)p \\
 &= B(z)(p - q) - B(z)p + B(y)q \\
 &= (B(y) - B(z))q
\end{align*}
\]

Thus, from Eqn. I we have:

\[
\|p - v\|_\infty \leq c_B \|y - z\|_\infty \|q\|_\infty \leq (c_B \|q\|_\infty )r
\]

In the algorithm, \( q = \bar{p}^t \) is the summation of estimations from all layers \( t = 1 \) to \( t - 1 \). Therefore:

\[
\|q\|_\infty = \|\bar{p}^t\|_\infty = \left\| \sum_{j=1}^{t-1} \bar{p}^j \right\|_\infty \leq \frac{r_1}{1 - \bar{\gamma}}
\]

and is thus bounded. Thus we have:

\[
\|H(I_p, q)(y) - I_{p-q}(y)\| = \|I_p(W(y; q)) - I_p(W(W^{-1}(y; p - q), p))\| = \|I_p(u) - I_p(v)\| \leq |\nabla I_p(\xi)||u - v||_{\infty}
\]

where \( \xi \in \text{Line} = \text{Seg}(u, v) \). Collecting Eqn. I over the entire region \( R \) gives the bound. When the algorithm runs on the distorted image \( I_p \), if the error during one iteration (when \( q = 0 \)) is included, the rectangle \( R \) moves from the initial location \( x, r \) to the final destination \( q = p \)

Practically the pull-back error \( \eta(x, r) \) is very small and can be neglected.

### 2.2 Relaxed Lipchitz Conditions

We put a generalized definition of relaxed Lipchitz Conditions here. The definition of relaxed Lipchitz conditions in our main paper is a special case for \( \eta(x, r) = 0 \).

**Assumption 3 (Relaxed Lipchitz Condition with pull-back error \( \eta(x, r) > 0 \))** There exists \( 0 < \alpha(x, r) \leq \gamma(x, r) < 1 \), \( A(x, r) > 0 \) and \( \Gamma(x, r) > A(x, r) + 2\eta(x, r) \) so that for any \( p_1 \) and \( p_2 \) with \( \|p_1\|_\infty \leq r, \|p_2\|_\infty \leq r \):

\[
\Delta p \leq \alpha r \implies \Delta I \leq Ar \quad (20)
\]

\[
\Delta p \geq \gamma r \implies \Delta I \geq \Gamma r \quad (21)
\]

for \( \Delta p \equiv \|p_1(S) - p_2(S)\|_\infty \) and \( \Delta I \equiv \|I_p(R) - I_{p_2}(R)\|_\infty \).

Here \( \|x\|_\infty \equiv \max \{x_i\} \). The error \( \eta(x, r) \) is from the property of pull-back operation (See Theorem I).

### 2.3 Guaranteed Nearest Neighbor

**Theorem 3 (Guaranteed Nearest Neighbor for Patch \( j \))** For any image patch \( (x, r) \), we have subset \( S = S(x, r) \) and image region \( R = R(x, r) \). Suppose we have a distorted image \( I \) so that \( \|I(R) - I_p(R)\| \leq \eta r \) with \( \|p\|_\infty \leq r \), then with

\[
\min \left( c_{SS} \left[ \frac{1}{\alpha} \right]^d, \left[ \frac{1}{\alpha} \right]^{2|S|} \right)
\]

number of samples properly distributed in the hypercube \([-r, r]^{2|S|}\), we can compute a prediction \( \hat{p}(S) \) so that

\[
|\hat{p}(S) - p(S)| \leq \gamma r
\]

using Nearest Neighbor in the region \( R \) with image metric. Here \( d \) is the effective degrees of freedom while \( 2|S| \) is the apparent degrees of freedom.
By the property of the relaxed Lipschitz conditions still holds no matter how many samples needed follows.

Thus, just setting the prediction \( \hat{p}(S) = q_{nn}(S) \) suffices.

**Theorem 4 (Verification of Aggregation Step.)** Suppose we have estimations \( \hat{p}(S_j) \) for overlapping \( S_j \) of the same layer covering the same landmark \( i \) (i.e., \( i \in S_j \)) so that the following condition holds:

\[
\| \hat{p}(S_j) - p(S_j) \|_\infty \leq r \quad \forall j
\]

Then the joint prediction

\[
\hat{p}(i) = \text{mean}_{j:i \in S_j} \hat{p}_{j-i}(S_j)
\]

satisfies \( \| \hat{p}(i) - p(i) \|_\infty \leq r \). As a result, \( \| \hat{p} - p \|_\infty \leq r \).

**Proof** By the property of \( \cdot \|_\infty \), we have for landmark \( i \):

\[
\| \hat{p}_{j-i}(S_j) - p(i) \|_\infty \leq r
\]

Then we have

\[
\| \hat{p}(i) - p(i) \| = \frac{1}{\# \{ j: i \in S_j \}} \sum_{j:i \in S_j} \| \hat{p}_{j-i}(S_j) - p(i) \| \leq r
\]
2.4 Number of Samples Needed

Theorem 5 (The Number of Samples Needed) The total number $N$ of samples needed is bounded by:

$$N \leq C_1 d_1^{d_1} + C_2 \log_{1/\bar{\gamma}} 1/\epsilon$$  \hspace{1cm} (35)

where $C_1 = 1/\min \alpha(x, r)$, $C_2 = 2^{1/(1-\bar{\gamma}^2)}$ and $C_3 = 2 + c_{SS}([\frac{1}{2} \log_{1/\bar{\gamma}} 2K/d] + 1)$.

Proof We divide our analysis into two cases: $d = 2K$ and $d < 2K$, where $K$ is the number of landmarks. $d > 2K$ is not possible. We index patch $(x, r)$ with subscript $j$, i.e., for $j$-th patch, its Lipschitz constants are $\alpha_j$, $\gamma_j$, $A_j$, $\Gamma_j$, etc. Besides, denote $[t]$ as the subset of all patches that belong to the same layer $t$.

Case 1: $d = 2K$

First let us consider the case that the intrinsic dimensionality of deformation field $d$ is just $2K$. Then the root dimensionality $d_1 = 2K$ (twice the number of landmarks). By Assumption $2$, the dimensionality $d_t$ for layer $t$ is:

$$d_t = \beta r_t^2 = \frac{d_1}{r_t^2} = \bar{\gamma}^{2t-2} d_1$$  \hspace{1cm} (36)

Any patch $j \in [t]$ has the same degrees of freedom since by Assumption $2$, $d_j$ only depends on $r_j$, which is constant over layer $t$.

For any patch $j \in [t]$, we use at most $N_j$ training samples:

$$N_j \leq \left( \frac{1}{\alpha_j} \right)^{d_t}$$  \hspace{1cm} (37)

to ensure the contracting factor is indeed at least $\gamma_j \leq \bar{\gamma}$. Note for patch $j$, we only need the content within the region $R_j$ as the training samples. Therefore, training samples of different patches in this layer can be stitched together, yielding samples that cover the entire image. For this reason, the number $N_t$ of training samples required for the layer $t$ is:

$$N_t \leq \arg \max_{j \in [t]} N_j \leq C_1^{d_1} = C_1^{\bar{\gamma}^{2t-2} d_1}$$  \hspace{1cm} (38)

for $C_1 = 1/\min \alpha_j$. Denote $n_t = C_1^{\bar{\gamma}^{2t-2} d_1}$. Then we have:

$$N \leq \sum_{t=1}^{T} N_t \leq \sum_{t=1}^{T} n_t$$  \hspace{1cm} (39)

To bound this, just cut the summation into half. Given $l > 1$, set $T_0$ so that

$$\frac{n_{T_0}}{n_{T_0+1}} = n_{T_0}^{1-\bar{\gamma}^2} \geq l, \quad \frac{n_{T_0+1}}{n_{T_0+2}} = n_{T_0+1}^{1-\bar{\gamma}^2} \leq l$$  \hspace{1cm} (40)

Thus we have

$$\sum_{t=1}^{T} n_t = \sum_{t=1}^{T_0} n_t + \sum_{t=T_0+1}^{T} n_t$$  \hspace{1cm} (41)

The first summation is bounded by a geometric series. Thus we have

$$\sum_{t=1}^{T_0} n_t \leq C_1^{d_1} \sum_{t=1}^{T_0} \left( \frac{1}{l} \right)^{t-1} \leq \frac{C_1^{d_1}}{1-1/l} = \frac{l}{l-1} C_1^{d_1}$$  \hspace{1cm} (42)

On the other hand, each item of the second summation is less than $l^{1/(1-\bar{\gamma}^2)}$. Thus we have:

$$\sum_{t=T_0+1}^{T} n_t \leq l^{1/(1-\bar{\gamma}^2)} T$$  \hspace{1cm} (43)
Combining the two, we then have:

\[ N \leq \frac{l}{l-1} C_1^{d_1} + \frac{1}{l-\gamma_2} T \]

for \( T = \lceil \log_{1/\gamma} 1/\epsilon \rceil \). Note this bound holds for any \( l \), e.g. \( 2 \). In this case, we have

\[ N \leq 2C_1^{d_1} + C_2 T \]

for \( C_2 = 2^{1/\gamma_2} \).

**Case 2:** \( d < 2k \)

In this case, setting \( d_1 = 2k \), finding \( T_1 \) so that \( d_{T_1} \geq d \) but \( d_{T_1+1} < d \) in Eqn. 36 yielding:

\[ T_1 = \left\lceil \frac{1}{2} \log_{1/\gamma} \frac{2K}{d} \right\rceil + 1 \]

Then, by Assumption 2, from layer 1 to layer \( T_1 \), their dimensionality is at most \( d \). For any layer between 1 and \( T_1 \), \( N_t \) is bounded by a constant number:

\[ N_t \leq c_{SS} C_1^d \]

The analysis of the layers from \( T_1 \) to \( T \) follow case 1, except that we have \( d \) as the starting dimension rather than \( 2k \). Thus, from Eqn. 45, the total number of samples needed is:

\[ N \leq (T_1 c_{SS} + 2)C_1^d + C_2 T \]

### 3 Sampling within a Hypercube

Theorem 3 is based on a design of sampling strategy so that for every location \( p \) in the hypercube \([-r, r]^D\), there exists at least one sample sufficiently close to it. Furthermore, we want to minimize the number of samples needed for this design. Mathematically, we want to find the smallest cover of \([-r, r]^D\).

In the following, we provide one necessary and two sufficient conditions. The first is for the general case (covering \([-r, r]^D\) entirely), while the second specifies the number of samples needed if \( p \) is known to be on a low-dimensional subspace, in which we could have better bounds.
3.1 Covering the Entire Hypercube

**Theorem 6 (Sampling Theorem, Necessary Conditions)** To cover $[-r, r]^D$ with smaller hypercubes of side $2\alpha r$ ($\alpha < 1$), at least $[1/\alpha^D]$ hypercubes are needed.

**Proof** The volume of $[-r, r]^D$ is $\text{Vol}(r) = (2r)^D$, while the volume of each hypercube of side $2\alpha r$ is $\text{Vol}(2\alpha r) = (2r)^D\alpha^D$. A necessary condition of covering is the total volume of small hypercubes has to be at least larger than $\text{Vol}(r)$:

$$N \text{Vol}(2\alpha r) \geq \text{Vol}(r)$$

which gives:

$$N \geq \frac{\text{Vol}(r)}{\text{Vol}(2\alpha r)} = \frac{1}{\alpha^D} \geq \left\lceil \frac{1}{\alpha^D} \right\rceil$$

\[ \blacksquare \]

**Theorem 7 (Sampling Theorem, Sufficient Conditions)** With $[1/\alpha]^D$ number of samples ($\alpha < 1$), for any $p$ contained in the hypercube $[-r, r]^D$, there exists at least one sample $\hat{p}$ so that $\|\hat{p} - p\|_\infty \leq \alpha r$.

**Proof** Uniformly distribute the training samples within the hypercube does the job. In particular, denote

$$n = \left\lceil \frac{1}{\alpha} \right\rceil$$

Thus we have $1/n = 1/[1/\alpha] \leq 1/(1/\alpha) = \alpha$. We put training sample of index $(i_1, i_2, \ldots, i_d)$ on $d$-dimensional coordinates:

$$\hat{p}_{i_1, i_2, \ldots, i_d} = r \left[-1 + \frac{2i_1 - 1}{n}, -1 + \frac{2i_2 - 1}{n}, \ldots, -1 + \frac{2i_D - 1}{n}\right]$$

does the job. Here $1 \leq i_k \leq n$ for $k = 1 \ldots D$. So each dimension we have $n$ training samples. Along the dimension, the first sample is $r/n$ distance away from $-r$, then the second sample is $2r/n$ distance to the first sample, until the last sample that is $r/n$ distance away from the boundary $r$. Then for any $p \in [-r, r]^D$, there exists $i_k$ so that

$$\left| p(k) - r \left(-1 + \frac{2i_k - 1}{n}\right) \right| \leq \frac{1}{n} r \leq \alpha r$$

This holds for $1 \leq k \leq D$. As a result, we have

$$\|p - \hat{p}_{i_1, i_2, \ldots, i_D}\|\infty \leq \alpha r$$

and the total number of samples needed is $n^D = [1/\alpha]^D$.

3.2 Covering a Subspace within Hypercube

Now we consider the case that $p$ lies on a subspace of dimension $d$, i.e., there exists a column-independent matrix $U$ of size $D$-by-$d$ so that $p = Uh$ for some hidden variable $h$. This happens if we use overcomplete local bases to represent the deformation. Since each landmark is related to two local bases, usually $D/2$ number of landmarks will give the deformation parameters $p$ with apparent dimension $D$.

In this case, we do not need to fill the entire hypercube $[-r, r]^D$. In fact, we expect the number of samples to be exponential with respect to only $d$ rather than $D$.

**Definition 8 (Noise Controlled Deformation Field)** A deformation field $p$ is called noise-controlled deformation of order $k$ and expanding factor $c$, if for every $p \in [-r, r]^D$, there exists a $k$-dimensional vector $(k \geq d) v \in [-r, r]^k$ so that $p = f(v)$. Furthermore, for any $v_1, v_2 \in [-r, r]^k$, we have:

$$\|p_1 - p_2\|\infty = \|f(v_1) - f(v_2)\|\infty \leq c\|v_1 - v_2\|\infty$$

for a constant $c \geq 1$. 

Note that by the definition of intrinsic dimensionality $d$, $v$ could be only $d$-dimensional and still $p = f(v)$. However, in this case, $c$ could be pretty large. In order to make $c$ smaller, we can have a redundant $k$-dimensional representation $h$ with $k > d$.

Many global deformation field satisfies Definition 8. Here we consider two cases, the affine deformation and the transformation that contains only translation and rotation.

**Affine transformation.** An affine deformation field $p$ defined on a grid has $d = 6$ and $k = 8$, no matter how many landmarks $(D/2)$ there are. This is because each component of $p$ can be written as

$$p(k) = [\lambda_1 x_k + \lambda_2 y_k + \lambda_3, \lambda_4 x_k + \lambda_5 y_k + \lambda_6]$$

for location $l_k = (x_k, y_k)$. Therefore, since any landmarks $l_k$ within a rectangle can be linearly represented by the locations of four corners in a convex manner, the deformation vector $p$ can also be linearly represented by the deformation vectors of four corners (8 DoF):

$$p(k) = A_k v = \sum_{j=1}^{4} a_{kj} v(j)$$

with $v$ is the concatenation of four deformation vectors from the four corners, $0 \leq a_{kj} \leq 1$ and $\sum_j a_{kj} = 1$. For any $p \in [-r, r]^D$, $v$ can be found by just picking the deformation of its four corners, and thus $\|v\|_\infty \leq r$. Furthermore, we have for $v_1, v_2 \in [-r, r]^k$:

$$\|p_1 - p_2\|_\infty = \|f(v_1) - f(v_2)\|_\infty \leq \max_k \sum_j a_{kj} |v_1(j) - v_2(j)| \leq \|v_1 - v_2\|_\infty$$

(58)

Therefore, $c = 1$.

**Transformation that contains only translation and rotation.** Similarly, for deformation that contains pure translation and rotation ($d = 3$), we just pick displacement vectors on two points ($k = 4$), the rotation center and the corner as $v$. Then we have:

$$p(r, \theta) = p_{\text{center}} + \frac{r}{r_{\text{corner}}} R(\theta)(p_{\text{corner}} - p_{\text{center}})$$

(59)

$$= (I - \frac{r}{r_{\text{corner}}} R(\theta)) p_{\text{center}} + \frac{r}{r_{\text{corner}}} R(\theta) p_{\text{corner}}$$

(60)

where $I$ is the identity matrix, $R(\theta)$ is the 2D rotational matrix and $r_{\text{corner}}$ is the distance from the center location to the corner. Here we reparameterize the landmarks with polar coordinates $(r, \theta)$. Therefore, for two different $v_1$ and $v_2$, since $r \leq r_{\text{corner}}$, we have:

$$\|p_1(r, \theta) - p_2(r, \theta)\|_\infty \leq \left\| (I - \frac{r}{r_{\text{corner}}} R(\theta))(p_{\text{center},1} - p_{\text{center},2}) \right\|_\infty$$

(61)

$$+ \left\| \frac{r}{r_{\text{corner}}} R(\theta)(p_{\text{corner},1} - p_{\text{corner},2}) \right\|_\infty$$

(62)

$$\leq 2 \|p_{\text{center},1} - p_{\text{center},2}\|_\infty + \sqrt{2} \|p_{\text{corner},1} - p_{\text{corner},2}\|_\infty$$

(63)

$$\leq (2 + \sqrt{2}) \|v_1 - v_2\|_\infty$$

(64)

since $|\cos(\theta)| + |\sin(\theta)| \leq \sqrt{2}$. Therefore,

$$\|p_1 - p_2\|_\infty \leq \max_{r, \theta} \|p_1(r, \theta) - p_2(r, \theta)\|_\infty \leq (2 + \sqrt{2}) \|v_1 - v_2\|_\infty$$

(65)

So $c = 2 + \sqrt{2} \leq 3.5$.

Given this definition, we thus have the following sampling theorem for deformation parameters $p$ lying on a subspace that is noise-controlled.

**Theorem 9 (Sampling Theorem, Sufficient Condition for Subspace Case)** For any noise-controlled deformation field $p = f(v)$ with order $k$ and expanding factor $c$, with $c_{SS}[1/\alpha]^d$ number of training samples distributed in the hypercube $[-r, r]^D$, there exists at least one sample $p$ so that $\|p - p\|_\infty \leq cr$. Note $c_{SS} = [c]^k \left[ \frac{1}{\alpha} \right]^{k-d}$. 9
We first apply Thm. 7 to the hypercube $[-r, r]^k$. Then with $\lceil \frac{c}{\alpha} \rceil^k$ samples, for any $\mathbf{v} \in [-r, r]^k$, there exists a training sample $\mathbf{v}'$ so that

$$\|\mathbf{v} - \mathbf{v}'\|_\infty \leq \frac{\alpha r}{c}$$ (66)

We then build the training samples $\{\mathbf{p}^i\}$ by setting $\mathbf{p}^i = f(\mathbf{v}^i)$. Therefore, from the definition of noise cancelling, given any $\mathbf{p} \in [-r, r]^D$, there exists an $\mathbf{v} \in [-r, r]^k$ so that $\mathbf{p} = f(\mathbf{v})$. By the sampling procedure, there exists $\mathbf{v}'$ so that $\|\mathbf{v} - \mathbf{v}'\|_\infty \leq \frac{\alpha r}{c}$, and therefore:

$$\|\mathbf{p} - \mathbf{p}'\|_\infty \leq c \|\mathbf{v} - \mathbf{v}'\|_\infty \leq \alpha r$$ (67)

setting $\hat{\mathbf{p}} = \mathbf{p}'$ thus does the job. Finally, note that

$$\left[ \frac{c}{\alpha} \right]^k \leq \left[ \frac{1}{\alpha} \right]^{k-d} \left[ \frac{1}{\alpha} \right]^d$$ (68)

So setting $c_{SS} = [c]_k \left[ \frac{1}{\alpha} \right]^{k-d}$ suffices (since $[ab] \leq [a][b]$).

4 Finding optimal curve $\gamma = \gamma(\alpha)$

Without loss of generality, we set $r = 1$. Then, we rephrase the algorithm in Alg. 1

**Algorithm 1** Find Local Lipschitz Constants

1: **INPUT** Parameter distances $\{\Delta \mathbf{p}_m\}$ with $\Delta \mathbf{p}_m \leq \Delta \mathbf{p}_{m+1}$.
2: **INPUT** Image distances $\{\Delta I_m\}$.
3: **INPUT** Scale $r$ and noise $\eta$.
4: $\Delta I^+_{m} = \max_{1 \leq l \leq m} \Delta I_l$, for $i = 1 \ldots M$.
5: $\Delta I^-_{m} = \min_{1 \leq l \leq M} \Delta I_l$, for $i = 1 \ldots M$.
6: for $m = 1$ to $M$ do
7: Find minimal $l^* = l^*(m)$ so that $\Delta I^-_{m} > \Delta I^+_{m} + 2\eta$.
8: if $m \leq l^*$ then
9: Store the 4-tuples $(\alpha, \gamma, A, \Gamma) = (\Delta \mathbf{p}_m, \Delta \mathbf{p}_{l^*}, \Delta I^+_{m}, \Delta I^-_{m})/r$.
10: end if
11: end for

To analyze Alg. 1 we make the following definitions:

**Definition 10 (Allowable set of $A$ and $\Gamma$)** Given $\alpha$, define the allowable set $\hat{A}(\alpha)$ as:

$$\hat{A}(\alpha) = \{ A : \forall m \Delta \mathbf{p}_m \leq \alpha \Rightarrow \Delta I_m \leq A \}$$ (69)

Naturally we have $\hat{A}(\alpha') \subset \hat{A}(\alpha)$ for $\alpha' > \alpha$. Similarly, given $\gamma$, define the allowable set $\hat{\Gamma}(\gamma)$ as:

$$\hat{\Gamma}(\gamma) = \{ \Gamma : \forall m \Delta \mathbf{p}_m \geq \gamma \Rightarrow \Delta I_m \geq \Gamma \}$$ (70)

and $\hat{\Gamma}(\gamma') \subset \hat{\Gamma}(\gamma)$ for $\gamma' < \gamma$.

**Lemma 11 (Properties of $\Delta I^+$ and $\Delta I^-$)** The two arrays constructed in Alg. 1 satisfy:

$$\Delta I^+_{m} = \min \hat{A}(\Delta \mathbf{p}_m)$$ (71)

$$\Delta I^-_{m} = \max \hat{\Gamma}(\Delta \mathbf{p}_m)$$ (72)

Moreover, $\Delta I^+_{m}$ is ascending while $\Delta I^-_{m}$ is descending with respect to $1 \leq m \leq M$.

**Proof (a):** First we show $\Delta I^+_{m} \in \hat{A}(\Delta \mathbf{p}_m)$. Since the list $\{\Delta \mathbf{p}_m\}$ was ordered, for any $\Delta \mathbf{p}_l \leq \Delta \mathbf{p}_{m-1}$, we have $l \leq m$. By definition of $\Delta I^+_{m}$, we have $\Delta I_l \leq \Delta I^-_{m}$. Thus $\Delta I^+_{m} \in \hat{A}(\Delta \mathbf{p}_m)$.

**Proof (b):** Then we show for any $A \in \hat{A}(\Delta \mathbf{p}_m)$, $\Delta I^-_{m} \leq A$. For any $1 \leq l \leq m$, since $\Delta \mathbf{p}_l \leq \Delta \mathbf{p}_m$, by the definition of $A$, we have $\Delta I_l \leq A$, and thus $\Delta I^-_{m} = \max_{1 \leq l \leq m} \Delta I_l \leq A$.

Therefore, $\Delta I^+_{m} = \min \hat{A}(\Delta \mathbf{p}_m)$. Similarly we can prove $\Delta I^-_{m} = \max \hat{\Gamma}(\Delta \mathbf{p}_m)$.
Theorem 12 For each $\alpha = \Delta p_m$, Algorithm[7] without the check $\alpha \leq \gamma$ always gives the globally optimal solution to the following linear programming:

\[
\begin{align*}
\text{min} & \quad \gamma \\
\text{s.t.} & \quad \Delta I_m \leq A \quad \forall \Delta p_m \leq \alpha \quad (\text{or} \quad A \in \tilde{A}(\alpha)) \\
 & \quad \Delta I_m \geq \Gamma \quad \forall \Delta p_m \geq \gamma \quad (\text{or} \quad \Gamma \in \tilde{\Gamma}(\gamma)) \\
 & \quad A + 2\eta < \Gamma 
\end{align*}
\]  

which has at least one feasible solution $(A \to +\infty, \gamma \to -\infty, \Gamma \to -\infty)$ for any $\alpha$.

Proof Since there are $M$ data points, we can discretize the values of $\alpha$ and $\gamma$ into $M$ possible values without changing the property of solution.

(a) First we prove every solution given by Alg. 1 (without the final check) is a feasible solution to the optimization (Eqn. 73). Indeed, for any $\alpha = \Delta p_m$, according to Lemma 11, $A = \Delta I_m^+ \in \tilde{A}(\alpha), \gamma = \Delta p_{l^*},$ and $\Gamma = \Delta I_{l^*} \in \tilde{\Gamma}(\gamma)$ and thus Eqn. 74 and Eqn. 75 are satisfied. From the construction of Alg. 1 $A + 2\eta < \Gamma$. Thus, the Algorithm 1 gives a feasible solution to Eqn. 73.

(b) Then we prove Alg. 1 (without the final check) gives the optimal solution. If there exists $l' < l^*$ so that $\gamma' = \Delta p_{l'} < \Delta p_{l^*} = \gamma$ is part of a better solution $(\alpha, \gamma', A', \Gamma')$, then $\tilde{\Gamma}(\gamma') \subset \tilde{\Gamma}(\gamma)$. This means

\[
A' + 2\eta < \Gamma' \leq \Delta I_{l'}^- = \max_{\tilde{\Gamma}(\gamma')} \leq \max_{\tilde{\Gamma}(\gamma)} = \Delta I_{l^*}^-. 
\]  

On the other hand, $A = \Delta I_m^+ = \min \tilde{A}(\alpha) \leq A' \in \tilde{A}(\alpha)$. Then, there are two cases:

- $\Delta I_m^+ + 2\eta < \Delta I_{l'}^- < \Delta I_{l^*}^-$. This is not possible since the algorithm already find the minimal $l^*$.

- $\Delta I_m^+ + 2\eta < \Delta I_{l'}^- = \Delta I_{l^*}^-$. Then according to the algorithm, $l' = l^*$.

which is a contradiction.

From Theorem 12 it is thus easy to check that the complete Algorithm[1] (with the check $\alpha \leq \gamma$) gives the optimal pair $(\alpha, \gamma)$ that satisfies the Relaxed Lipschitz Conditions (Eqn. 20 and Eqn. 21).

5 More Experiments

Fig. 4 shows the behaviors of our algorithm over different iterations. We can see with more and more stages, the estimation captures more detailed structures and becomes better.

Fig. 5 shows how the performance degrades if only the bottom $K$ layers are used for prediction. We can see that each layer plays a different rule. Layer 3-4 seems to be critical for the synthetic data since they have captured the major mode/scale of deformation.

References

<table>
<thead>
<tr>
<th>Test Image</th>
<th>Initialization</th>
<th>Iteration 1</th>
<th>Iteration 3</th>
<th>Iteration 5</th>
<th>Final Result</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>10.934074</td>
<td>9.694556</td>
<td>6.870912</td>
<td>4.065068</td>
<td>2.555294</td>
</tr>
<tr>
<td></td>
<td>10.830096</td>
<td>8.749503</td>
<td>6.417697</td>
<td>4.283149</td>
<td>2.577389</td>
</tr>
<tr>
<td></td>
<td>12.664164</td>
<td>9.052568</td>
<td>7.401009</td>
<td>5.070467</td>
<td>3.802439</td>
</tr>
<tr>
<td></td>
<td>8.968721</td>
<td>12.248940</td>
<td>5.515930</td>
<td>3.479620</td>
<td>2.133142</td>
</tr>
<tr>
<td></td>
<td>9.700289</td>
<td>8.678155</td>
<td>7.584708</td>
<td>4.101575</td>
<td>2.587239</td>
</tr>
<tr>
<td></td>
<td>11.256226</td>
<td>10.846395</td>
<td>7.610691</td>
<td>5.212512</td>
<td>3.858836</td>
</tr>
</tbody>
</table>

Figure 4: Landmark Estimation at different iterations given by our approach.
Figure 5: Landmark Estimation using only last $L$ layers of the hierarchy. Layer 3-4 is critical for getting a good estimation of the landmarks on the synthetic data.