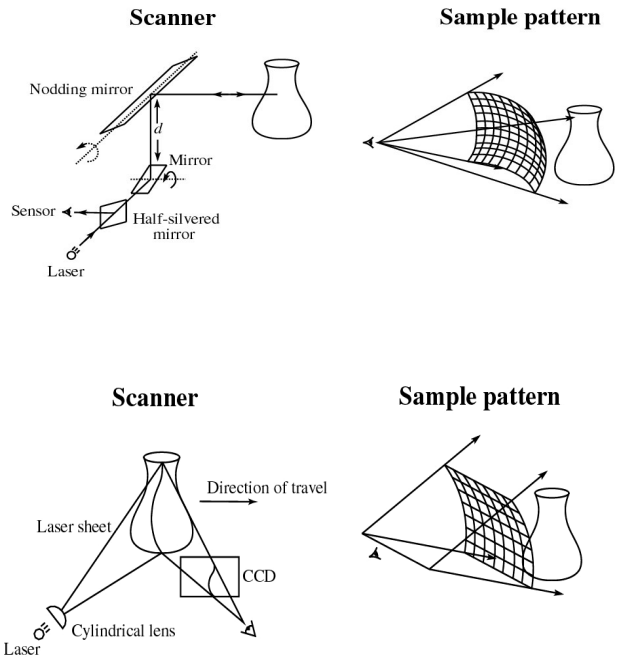
Range image registration

Range images

For many structured light scanners, the range data forms a highly regular pattern known as a range image.

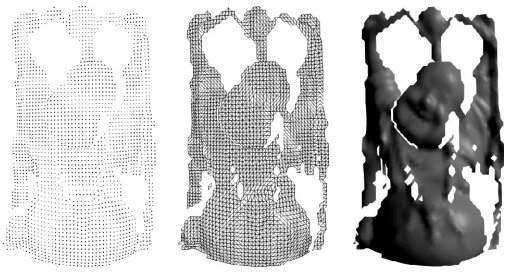
The sampling pattern is determined by the specific scanner.

Examples of sampling patterns



Range images and range surfaces

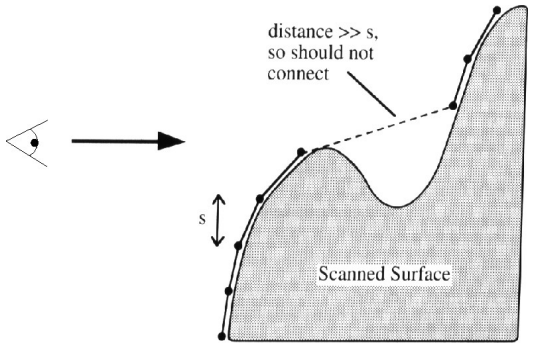
Given a range image, we can perform a preliminary reconstruction known as a range surface.



Range image Tesellation Range surface

Tessellation threshold

To avoid "prematurely aggressive" reconstruction, a tessellation threshold is employed:



Registration

Any surface reconstruction algorithm strives to use all of the detail in the range data.

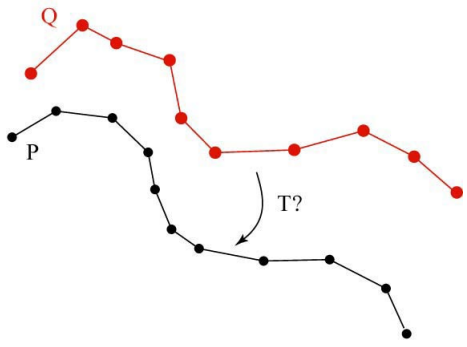
To preserve this detail, the range data must be precisely registered.

Accurate registration may require:

- Calibrated scanner positioning
- Software optimization
- Both

Registration

Problem: given two overlapping range scans, what is the rigid transformation, **T**, that minimizes the distance between them.



Least squares error

How do we measure this distance?

If we think of surfaces, we can pose a least squares problem in integral form, something like:

$$E = \int \left\| \mathbf{p}(u,v) - \mathbf{T}\mathbf{q}(u,v) \right\|^2 du dv$$

where **p**(u,v) and **q**(u,v) are corresponding points on **P** and **Q**, respectively.

Alternatively, we can write out a sampled version of this:

$$E = \sum_{i}^{N} \left\| \mathbf{p}_i - \mathbf{T}\mathbf{q}_i \right\|^2$$

where $\mathbf{p}_i$ and $\mathbf{q}_i$ are corresponding samples on **P** and **Q**, respectively.

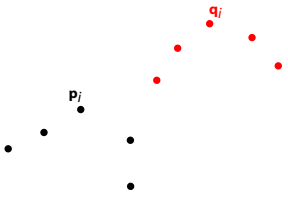
Solution to least squares problem

A derivation due to Horn shows that there is a closed form solution to the problem of finding the T that minimizes:

$$E = \sum_i^N \left\| \mathbf{p}_i - \mathbf{T}\mathbf{q}_i \right\|^2$$

This solution is for the class of T's that permit scale, rotation, and translation. We'll just allow the latter two (rigid body transformations):

$$E = \sum_i^N \left\| \mathbf{p}_i - (\mathbf{R}\mathbf{q}_i + \mathbf{t}) \right\|^2$$



Solution to least squares problem

To solve, we first compute the centroid of each point set:

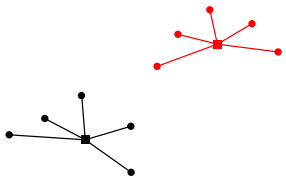
$$\bar{\mathbf{p}} = \sum_i^N \mathbf{p}_i \quad \bar{\mathbf{q}} = \sum_i^N \mathbf{q}_i$$

Horn showed that the best rotation satisfies:

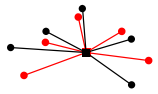
$$\underset{\mathbf{R}}{\operatorname{argmax}} \sum_i^N (\mathbf{p}_i - \bar{\mathbf{p}})^T \mathbf{R} (\mathbf{q}_i - \bar{\mathbf{q}})$$

In other words:

1. Convert the points into vectors relative to their centroids.
2. Find a rotation that makes corresponding vectors have dot products as close to 1 as possible.



Centroid relative



Common origin

Solution to least squares problem

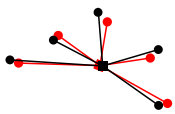
To solve for this rotation, you can construct a 3x3 matrix:

$$\mathbf{M} = \sum_i^N (\mathbf{p}_i - \bar{\mathbf{p}})(\mathbf{q}_i - \bar{\mathbf{q}})^T$$

and then solve:

$$\mathbf{R} = \mathbf{M}(\mathbf{M}^T\mathbf{M})^{-1/2}$$

which amounts to solving an eigenvalue problem for a 3x3 matrix.



The optimal translation is then just:

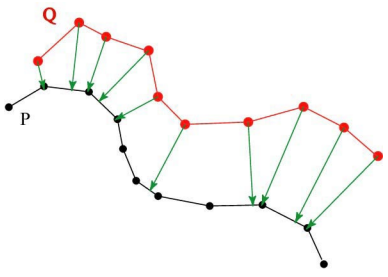
$$\mathbf{t} = \bar{\mathbf{p}} - \mathbf{R}\bar{\mathbf{q}}$$

Correspondences: "closest" points

So, we now have a closed form solution for **T** given corresponding $\mathbf{p}_i$ and $\mathbf{q}_i$.

How do we get these correspondences??

One solution is to find the *nearest* points to $\mathbf{q}_i$ that lie on **P**.



Note that the resulting $\mathbf{p}_i$ can lie on faces, edges, and vertices of **P**.

Alternatives include nearest point:

- along the direction of the normal at $\mathbf{q}_i$
- along a fixed direction

Iterated Closest Point (ICP)

After finding the best T based on these "correspondences," we will have brought the surfaces closer together, but not all the way.

How do we go the rest of the way? Iterate!

until E is small
    Identify nearest points
    Compute the optimal **T**
end until

This procedure, called **Iterated Closest Point (ICP)**, was developed by Besl and McKay.

Q: What kinds of practical problems do you think you will encounter when aligning two range scans?

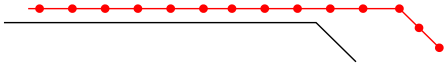
Springs that slow convergence

One shortcoming of the ICP method is slow convergence.

We can think of a least squares solution as:

1. Tacking a bunch of springs between points
2. Requiring their rest lengths to be zero
3. Solving for the lowest energy configuration

If many of the points are near each other, but should slide past each other, the springs will resist:



Q: how might you speed this up?

"Sliding" springs

Chen and Medioni proposed an alternate error function that does not penalize sliding.

In particular, at each closest point, $\mathbf{p}_i$, the normal defines a tangent plane:

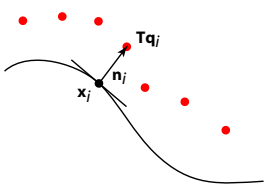
$$\mathbf{n}_i^T(\mathbf{x}-\mathbf{p}_i)=0$$

Signed distance from this plane is simply:

$$d(\mathbf{x})=\mathbf{n}_i^T(\mathbf{x}-\mathbf{p}_i)$$

The error function can now be written in terms of square distances from planes:

$$E=\sum_i^N\left[\mathbf{n}_i^T(\mathbf{T}\mathbf{q}_i-\mathbf{p}_i)\right]^2$$



"Sliding" springs

There is no known closed solution for T in this case, but it can be solved quickly in a few linear sub-iterations.

The algorithm otherwise proceeds as ICP.

Result: faster convergence.

This was the method of choice for the Digital Michelangelo Project.

Error accumulation

Consider a set of N scans around an object.

With each pairwise registration you get a least squares optimal transformation.

Will this transformation bring the range data into perfect alignment?

What happens when you come full circle and compare scan N-1 to scan 0?

Global registration

The problem now becomes: find the set of transformations that *simultaneously* minimizes distances between range scans.

This is sometimes called the **global registration** problem.

One solution is to define a new global error function and solve for the best $\mathbf{T}_j$ in:

$$E = \sum_{j}^{M} \sum_{k}^{M} \sum_{i}^{N_{jk}} \left\| \mathbf{T}_j \mathbf{p}_{ji} - \mathbf{T}_k \mathbf{p}_{ki} \right\|^2$$

where:

- M is the number of scans
- N_{jk} is the number of points in correspondence between scans j and k
- $\mathbf{T}_j$ is the transformation for scan j
- $\mathbf{p}_{ji}$ is the i-th point from the j-th scan

Can initialize with pairwise ICP and then perform a *large, global, non-linear* ICP.

Global registration

For the Digital Michelangelo Project, Kari Pulli developed a simpler, faster version of global ICP.

One suggested approach:

Perform pairwise registration.
Save a sub-sampled "best" set of pairwise correspondences.
until convergence
Select next scan j
Compute the optimal $\mathbf{T}_j$ w.r.t. E_j:

$$E_j = \sum_{k}^{M} \sum_{i}^{N_{jk}} \left\| \mathbf{T}_j \mathbf{p}_{ji} - \mathbf{T}_k \mathbf{p}_{ki} \right\|^2$$

end until

Global registration

Pulli modifies this to:

- keep the original pairwise transforms, $\mathbf{T}_{j \to k}$
- substitute $\mathbf{p}_{ki}$ with $\mathbf{T}_{j \to k}\, \mathbf{p}_{ji}$

The error function at each step is then:

$$E_j = \sum_{k}^{M} \sum_{i}^{N_{jk}} \left\| \mathbf{T}_j \mathbf{p}_{ji} - \mathbf{T}_k \mathbf{T}_{j \to k} \mathbf{p}_{ji} \right\|^2$$

Non-rigid registration

Calibrating scanners can be extremely difficult.

The DMP scanner was not 100% calibrated. How to compensate?

Solution: fold non-linear scanner parameters into some of the registration procedures.

Q: Is there an analagous problem in computer vision?

Bibliography

Besl, P.J. and McKay, H.D., "A method for registration of 3-D shapes," IEEE Transactions on Pattern Analysis and Machine Intelligence, Feb. 1992, 2(4), pp. 239--256.

Chen, Y. and Medioni, G., "Object modeling by registration of multiple range images," Image and Vision Computing, 10(3), April, 1992, pp. 145-155.

Horn BKP, Hilden HM, Negahdaripour S, "Closed-form Solution of Absolute Orientation Using Orthonormal Matrices," Journal of the Optical Society of America, Series A, 5, 7, 1988, pp 1127-1135.

K. Pulli. "Multiview Registration for Large Data Sets," Int.Conf. on 3D Digital Imaging and Modeling, Ottawa, pp.160-168, 1999