Michael Van Maele, Group #3
24 April 2012
EN.600.646 – CIS2

Seminar Article Report:
Rigid and Articulated Point Registration with Expectation Conditional Maximization
Horaud R, Forbes F, Yguel M, Dewaele G, Zhang JA

RELEVANCE, INTENT & IMPORTANCE

Relevance

The selection of this particular article can be justified for numerous reasons. Firstly, the work represents a cutting-edge contribution to 3D/3D pose estimation. Not only is this a preponderant topic in the field of point registration, but it is also a problem strikingly similar to the one that must be solved in the current project. [1]

Secondly, the key result of the work is a variation on the ECM algorithm. A previously coded implementation of the ECM algorithm is employed in the current project. The dissection and analysis of this article has afforded invaluable insight into the theory and practice of the application of EM-like algorithms. The deliverables of the current project depend on a thorough understanding of the inner workings of methods based on those presented in this article.

Thirdly, the methods outlined in the article and those used in the current project have the following implementation parallels:

1. The use of an ECM approach to performing maximum-likelihood estimation (MLE).
2. The use of a Gaussian Mixture Model (GMM).
3. The use of a uniform distribution to handle outliers.

NOTE: The authors proposed both a method for rigid point registration and articulated point registration. Being that the latter makes many calls to the former, and also given that the primary interest in this work in the context of the current project lies in the rigid registration aspect, less attention is given to the articulated algorithm details and figures (for example, Figure 3 on page 595 of the original article is not elaborated). Also, most of the equations in this report consist of mathematical steps left out by the authors but which have been provided here for the sake of clarity. Any diagrams (particularly flow charts) contained in this report which are not credited to the original source are the work of the author of this report.

Intent

In this investigation, the authors cast the point registration problem into and MLE framework with missing data. The intent of the authors was fourfold:

1. To formally derive a novel, ECM-based algorithm for rigid point registration.
2. To elaborate on the practical considerations of the implementation of the novel algorithm.
3. To extend the application of the novel algorithm from rigid point registration to articulated point registration.

[1] http://ciis.lcsr.jhu.edu/dokuwiki/doku.php?id=courses:446:2012:446-2012-03:project03

4. To compare the performance of the novel algorithm to that of the leading alternative algorithm.

Importance

The authors claim that an ECM-based point registration (ECMPR) algorithm is of value primarily because it is 1) more broadly applicable than other EM-like point registration methods and 2) does not disregard other model parameters when maximizing over the registration parameters. That is to say, the maximization of the registration is *conditional* on the other model parameters.

Moreover, the authors demonstrate that their point registration algorithm is robust to outliers. The technique used by the authors for the rejection of outliers has advantages over existing methods, including the tendency to avoid becoming trapped at local minima (thanks to the EM structure of the algorithm) and efficiency.

By all accounts, the authors succeeded in their undertaking. However, certain shortcomings do exist, and these will be addressed near the close of this report. What follows is an overview of the technical approach chosen by the authors, the experiments designed by the authors, the results of those experiments, and the conclusions that they authors have drawn from them.

TECHNICAL SUMMARY

This technical summary will begin by providing the reader with details of the mathematical notation used in the article. It will continue by defining the problem at hand, and it will then outline the rigid point registration algorithm (named ECMPR) proffered by the authors. Next, the report will "backtrack" through the key steps of the formal derivation of the ECMPR-Rigid algorithm. Finally, the technical summary will describe the extension of the ECMPR-Rigid algorithm to articulated registration (ECMPR-Articulated). Justifications for the manipulations of various equations which may not have been clearly stated by the authors will also be provided.

Notation

Although the mathematical notation used in the article is explicitly defined by the authors, a more thorough table of mathematical definitions is desirable for a "first-pass" reading. Such a guide is presented on the final page of this document. The reader may wish to detach that page for reference before continuing.

Problem Statement

Point Registration

The *point registration* (PR) problem seeks to determine the best possible alignment between two sets of points. It consists of two main steps. Firstly, the point-to-point correspondences must be obtained. Secondly, the transformation mapping one set of points onto the other set of points must be estimated. In the rigid PR case, this transformation consists of two parts: a component of *rotation* $\mathbf{R}$ and a component of *translation* $\boldsymbol{t}$. The authors utilize an *expectation-maximization (EM) framework* as the jumping point for their work.

When the PR problem is cast into the EM framework, a maximization criterion must be defined from which the optimal model parameters can be estimated. That is, an *objective* function must be chosen, and it must be maximized (or minimized, as the case may be) over the model parameters. Thus, it can be called a *maximum-likelihood estimator.* The optimal parameters are those which take the maximum-likelihood estimator to its extreme value. However, the fact that data is missing from the problem (specifically, the correspondences between points) complicates maximization. There are several approaches to handling such hidden information problems (for example, fitting a Hidden Markov Models). The authors have chosen to fit a *Gaussian mixture model* (GMM) to deal with hidden information.

Gaussian Mixture Models

In this investigation, the authors fit a GMM to the observed data $\mathcal{Y}$ by assuming that the transformed model points, denoted $\boldsymbol{\mu}(\boldsymbol{X}_i; \boldsymbol{\Theta})$, coincide with the means of a mixture of Gaussian densities. The observed data points $\boldsymbol{Y}_i$ are assumed to be *random variables*, or observations drawn from these distributions. The probability that $\boldsymbol{Y}_i$ actually is such an observation is computable. An observed point very distant from a model point for whose Gaussian density it is being considered a putative random variable will have a weak probability of having been drawn from that density. This is clear later when the posterior probabilities and model parameters are shown to have distance terms in them. Once the maximum-likelihood estimator is defined and expanded, it will become clear that these probabilities (as well as others) will have to be defined in order to describe the ECMPR-Rigid algorithm. The full algorithm is summarized now for reference, and the steps taken to derive the algorithm will then be described.

The ECMPR-Rigid Algorithm

1. *Initialization:* Set $\mathbf{R}^q = \mathbf{I}$ and $\boldsymbol{t}^q = \vec{\mathbf{0}}$. Choose the initial covariance matrices $\boldsymbol{\Sigma}_i^q,\ i = 1 \dots n$.
2. *E-step*: Evaluate the posteriors α_{ji}^q, the virtual observation $\boldsymbol{W}_i^q$, and its weight λ_i^q using the expressions provided below with the <u>current</u> parameters $\mathbf{R}^q$, $\boldsymbol{t}^q$, and $\boldsymbol{\Sigma}_i^q$.
3. *CM-Steps*:
   a. Use *semidefinite positive* (SDP) relaxation to estimate the new rotation matrix $\mathbf{R}^{q+1}$ by minimization of the expression provided below for $\mathbf{R}^*$ with α_{ji}^q and $\boldsymbol{W}_i^q,\ i = 1 \dots n$.
   b. Estimate the new translation vector $\boldsymbol{t}^{q+1}$ using the expression provided below for $\boldsymbol{t}^*$ with the <u>new</u> rotation matrix $\mathbf{R}^{q+1}$ and the <u>current</u> posteriors α_{ji}^q and the <u>current</u> covariances $\boldsymbol{\Sigma}_i^q,\ i = 1 \dots n$.
   c. Estimate the <u>new</u> covariances $\boldsymbol{\Sigma}_i^{q+1},\ i = 1 \dots n$ using the expressions provided below with the <u>current</u> posteriors α_{ji}^q, the <u>new</u> rotation matrix $\mathbf{R}^{q+1}$, and the <u>new</u> translation vector $\boldsymbol{t}^q$.
4. *Convergence:* Compare the new and current rotation matrices, $\mathbf{R}^q$ and $\mathbf{R}^{q+1}$. If $\|\mathbf{R}^q - \mathbf{R}^{q+1}\|^2 < \epsilon$ (where ϵ is a threshold), then go to the *Classification* step.
5. *Classification*: Assign each observation to a model point (inlier) or to the uniform class (outlier) based on the maximum a posteriori (MAP) principle:

   $$z_j = \arg\max_i \alpha_{ji}^q$$

   That is, let the index of the model point $\boldsymbol{X}_i$ corresponding to the observed data point $\boldsymbol{Y}_j$ be equal to the value of i which maximizes the value of the posterior, α_{ji}^q.

The ECMPR-Rigid algorithm depends on the iterated computation of various quantities, including the model parameters $\boldsymbol{\Theta} \coloneqq \{\mathbf{R}, \boldsymbol{t}, \boldsymbol{\Sigma}_i \text{ for } i = 1 \dots n\}$, the inlier and outlier posteriors, and more. The highlights of the formal derivations of these key quantities will now be reviewed, beginning with the posterior probabilities.

Posterior Probabilities ($\alpha_{ji}, i = 1 \dots n$)

The posteriors $\alpha_{ji}, i = 1 \dots n$ are arguably the most difficult quantities to accept in this work. They are defined as the likelihood of the assignment of observed data point $\boldsymbol{Y}_j$ to model point $\boldsymbol{X}_i$ given the observation of $\boldsymbol{Y}_j$. This treatment will attempt to demystify both the derivation and significance of these quantities.

Bayes' Rule

The derivation of the posterior probabilities begins with Bayes' rule. In general, it is stated as

$$P(A = a|b) = \frac{P(b|A = a) \cdot P(A = a)}{P(b)}$$

BAYES' RULE

Where $p(A = a|b)$ is the probability that random variable A takes on the value a given condition b. In plain English, this is interpreted as

$$\text{posterior} = \frac{\text{likelihood} \cdot \text{prior}}{\text{marginal likelihood}}$$

BAYES' RULE

Note that the marginal likelihood is simply a *normalizing constant*. That is, it ensures that the sum of the posterior over all possible values of A is equal to 1. To adapt Bayes' rule to the language used in the article, the formula is rewritten as

$$P(Z_j = i|\boldsymbol{Y}_j) = \frac{P(\boldsymbol{Y}_j|Z_j = i) \cdot P(Z_j = i)}{P(\boldsymbol{Y}_j)} = \alpha_{ji}$$

BAYES' RULE

Checking the correspondences among the two instantiations of Bayes' rule will show that the two formulas are completely analogous.

Priors

Note that the prior in this case is $P(Z_j = i)$. The authors state that in the EM framework, the priors are treated as model parameters. To parameterize the priors, they envision a spherical volume about each model point $\boldsymbol{X}_i$ which constitutes a fraction of the entire 3D working volume V. The priors are labeled $p_i = P(Z_j = i)$ and are parameterized according to

$$p_i = \text{priors} = P(Z_j = i) = \begin{cases} p_{in} = \dfrac{v}{V}, & \text{if } 1 \leq i \leq n \\ p_{out} = \dfrac{V - nv}{V}, & \text{if } i = n + 1 \end{cases}$$

PARAMETERIZATION OF PRIORS

where $v = 4\pi r^3/3$ is the volume of said sphere of radius r, centered at a model point $\boldsymbol{X}_i$. It is assumed that $nv \ll V$. In designing this parameterization of the priors, the authors are providing *prior information* to the model about the missing information. In other words, if a point is an inlier (which is true if the index of the matching model point is within $[1, n]$), then $P(Z_j = i)$ will be believed to be an inlier with prior certainty $\frac{v}{V}$. On the other hand, if the index of the matching model point is $n+1$, then $P(Z_j = i)$ will be believed to be an outlier with a prior certainty of ≈ 1.

Now that the *prior* component of Bayes' rule has been defined, the *likelihood* and the *marginal likelihood* remain.

Likelihood

Recall that the *likelihood* component of the posterior is written as $P(\boldsymbol{Y}_j | Z_j = i)$. In the language of the article, this quantity is the likelihood of an observation point $\boldsymbol{Y}_j$, given that the missing information Z_j suggests $\boldsymbol{Y}_j$ corresponds to the model point $\boldsymbol{X}_i$. The choice of the distribution from which to draw the likelihood is somewhat of an art form. The authors seek to fit a GMM to the observed data points such that the centers of the Gaussian densities lie on the transformed model points, $\boldsymbol{\mu}(\boldsymbol{X}_i; \boldsymbol{\Theta}), \boldsymbol{X}_i \in \mathcal{X}$. Therefore, the likelihood values will be drawn from a Gaussian distribution.

In general the Gaussian distribution may be written as

$$f(x; \mu, \sigma^2) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left[-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2\right]$$

NORMAL (GAUSSIAN) DISTRIBUTION

where μ is the mean and σ^2 is the variance. However, in this work, the distribution is *multivariate*. Therefore, the applicable formula is

$$\mathbf{x} \sim \mathcal{N}_k(\boldsymbol{\mu}, \boldsymbol{\Sigma})$$

k−DIMENSIONAL GAUSSIAN DISTRBIUTION

where $\boldsymbol{\mu}$ is the vector of means and $\boldsymbol{\Sigma}$ is the covariance matrix. Note that the covariance matrix generalizes the concept of variance to three dimensions. The article does not explicitly write the formula for the 3-dimensional Gaussian distribution. However, it is supplied here for reference:

$$f_{\mathbf{x}}(x_1, x_2, x_3) = \left[(2\pi)^{3/2}|\Sigma|^{1/2}\right]^{-1} \exp\left[-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^{\mathrm{T}}\Sigma^{-1}(\mathbf{x}-\boldsymbol{\mu})\right]$$

where $|\Sigma|$ is the determinant of the covariance matrix. Thus, the likelihoods can be written as

$$\text{likelihoods} = P(\boldsymbol{Y}_j | Z_j = i) = \mathcal{N}(\boldsymbol{Y}_j |\, \boldsymbol{\mu}(\boldsymbol{X}_i; \boldsymbol{\Theta}), \boldsymbol{\Sigma}_i), \qquad \text{for } 1 \le i \le n$$

Note that the likelihoods have only been found for $1 \le i \le n$. The authors chose to draw the likelihoods for $i = n+1$ from a *uniform distribution*, which may generally be written as

$$\mathcal{U}(a,b) = \begin{cases} \frac{1}{b-a}, & \text{for } x \in [a,b] \\ 0, & \text{otherwise} \end{cases}$$

UNIFORM DISTRIBUTION

However, the authors have made a notational error. They write

$$\mathcal{U}(\boldsymbol{Y}_j | V, 0)$$

when they actually mean

$$\mathcal{U}(\boldsymbol{Y}_j|0,V)$$

In any case, the likelihood of an observation given its assignment to the outlier cluster is drawn from the uniform distribution over the entire 3D working volume. Thus, we have

$$\text{likelihoods} = P(\boldsymbol{Y}_j|Z_j = n+1) = \mathcal{U}(\boldsymbol{Y}_j|0,V) = \frac{1}{V}$$

All that remains is the *marginal likelihood.*

Marginal Likelihood

Recall that the *marginal likelihood* is the denominator of the posterior. As the normalization factor, it consists of the sum of all possible values that the numerator can take on. Remember that

$$\text{marginal likelihood} = P(\boldsymbol{Y}_j)$$

$$P(\boldsymbol{Y}_j) = \sum_{i=1}^{n+1} P(\boldsymbol{Y}_j|Z_j = i)\, P(Z_j = i)$$

$$P(\boldsymbol{Y}_j) = \sum_{i=1}^{n+1} P(\boldsymbol{Y}_j|Z_j = i)\, p_i$$

The authors report this equation as the complete expression for the marginal likelihood.

Final Result (α_{ji}^q)

Substitution of these components into Bayes' rule gives

$$\text{posterior} = \frac{\text{likelihood} \cdot \text{prior}}{\text{marginal likelihood}}$$

$$\alpha_{ji} = \frac{\mathcal{N}\left(\boldsymbol{Y}_j|\, \boldsymbol{\mu}(\boldsymbol{X}_i;\boldsymbol{\Theta}),\boldsymbol{\Sigma}_i\right)p_i}{\sum_{i=1}^{n+1}\mathcal{N}\left(\boldsymbol{Y}_j|\, \boldsymbol{\mu}(\boldsymbol{X}_i;\boldsymbol{\Theta}),\boldsymbol{\Sigma}_i\right)p_k}$$

The numerator and the denominator differ only in the presence of the summation in the denominator. Expanding the $\mathcal{N}(\cdot)$ function notation gives

$$\alpha_{ji} = \frac{\left[(2\pi)^{3/2}|\boldsymbol{\Sigma}_i|^{1/2}\right]^{-1}\exp\left[-\frac{1}{2}\left(\boldsymbol{Y}_j - \boldsymbol{\mu}(\boldsymbol{X}_i;\boldsymbol{\Theta})\right)^{\mathrm{T}}\boldsymbol{\Sigma}_i{}^{-1}\left(\boldsymbol{Y}_j - \boldsymbol{\mu}(\boldsymbol{X}_i;\boldsymbol{\Theta})\right)\right]p_i}{\sum_{k=1}^{n+1}[(2\pi)^{3/2}|\boldsymbol{\Sigma}_k|^{1/2}]^{-1}\exp\left[-\frac{1}{2}\left(\boldsymbol{Y}_j - \boldsymbol{\mu}(\boldsymbol{X}_k;\boldsymbol{\Theta})\right)^{\mathrm{T}}\boldsymbol{\Sigma}_k{}^{-1}\left(\boldsymbol{Y}_j - \boldsymbol{\mu}(\boldsymbol{X}_k;\boldsymbol{\Theta})\right)\right]p_k}$$

Canceling the constants gives

$$\alpha_{ji} = \frac{|\mathbf{\Sigma}_i|^{-1/2} \exp\left[-\frac{1}{2}\left(\boldsymbol{Y}_j - \boldsymbol{\mu}(\boldsymbol{X}_i;\mathbf{\Theta})\right)^{\mathrm{T}} \mathbf{\Sigma}_i{}^{-1}\left(\boldsymbol{Y}_j - \boldsymbol{\mu}(\boldsymbol{X}_i;\mathbf{\Theta})\right)\right] p_i}{\sum_{k=1}^{n+1} |\mathbf{\Sigma}_i|^{-1/2} \exp\left[-\frac{1}{2}\left(\boldsymbol{Y}_j - \boldsymbol{\mu}(\boldsymbol{X}_k;\mathbf{\Theta})\right)^{\mathrm{T}} \mathbf{\Sigma}_k{}^{-1}\left(\boldsymbol{Y}_j - \boldsymbol{\mu}(\boldsymbol{X}_k;\mathbf{\Theta})\right)\right] p_k}$$

The authors give an expression for the quantity known as the *Mahalanobis distance*,

$$\|\boldsymbol{X} - \boldsymbol{Y}\|_{\Sigma}^2 = (\boldsymbol{X} - \boldsymbol{Y})^{\mathrm{T}} \mathbf{\Sigma}^{-1} (\boldsymbol{X} - \boldsymbol{Y})$$

which allows the above to be simplified. Rewritten, the posterior is

$$\alpha_{ji} = \frac{|\mathbf{\Sigma}_i|^{-1/2} \exp\left(-\frac{1}{2}\left\|\boldsymbol{Y}_j - \boldsymbol{\mu}(\boldsymbol{X}_i;\mathbf{\Theta})\right\|_{\Sigma_i}^2\right) p_i}{\sum_{k=1}^{n+1} |\mathbf{\Sigma}_i|^{-1/2} \exp\left(-\frac{1}{2}\left\|\boldsymbol{Y}_j - \boldsymbol{\mu}(\boldsymbol{X}_i;\mathbf{\Theta})\right\|_{\Sigma_k}^2\right) p_k}$$

Substituting for p_i and p_k gives

$$\boxed{\alpha_{ji} = \frac{|\mathbf{\Sigma}_i|^{-1/2} \exp\left(-\frac{1}{2}\left\|\boldsymbol{Y}_j - \boldsymbol{\mu}(\boldsymbol{X}_i;\mathbf{\Theta})\right\|_{\Sigma_i}^2\right)}{\sum_{k=1}^{n+1} |\mathbf{\Sigma}_i|^{-1/2} \exp\left(-\frac{1}{2}\left\|\boldsymbol{Y}_j - \boldsymbol{\mu}(\boldsymbol{X}_i;\mathbf{\Theta})\right\|_{\Sigma_k}^2\right) + \emptyset_{3D}}}$$

Horaud et al., Equation (12), page 591

where $\emptyset_{3D}$ accounts for outliers and is equal to $\frac{r^3}{1.5\sqrt{2\pi}}$.

All that is needed is $\alpha_{j\,n+1}$. This is easily defined as one minus the total posterior probability of an assignment to an inlier, or

$$\boxed{\alpha_{j\,n+1} = 1 - \sum_{i=1}^{n} \alpha_{ji}}$$

Horaud et al., Equation (14), page 591

Next, the procedure for obtaining the maximum-likelihood estimator of the model parameters is described.

Maximum-Likelihood Estimator

In the EM framework, one seeks to find by iteration the *maximum-likelihood estimate* of the model parameters. That is, one searches for the assignment of the parameter set $\mathbf{\Psi} = \{\mathbf{\Theta}, \mathbf{\Sigma_1}, \dots, \mathbf{\Sigma}_n\}$ which maximizes the value of some objective function, representing a point-to-point alignment *no worse* than the initial values of the parameter set. Ordinarily, the *observed-data log-likelihood* would serve as the objective function to maximize, written as

$$\mathcal{L}(\mathbf{\Theta}, \mathbf{\Sigma_1}, \dots, \mathbf{\Sigma}_n | \mathcal{Y}) = \log P(\mathcal{Y}; \mathbf{\Theta}, \mathbf{\Sigma_1}, \dots, \mathbf{\Sigma}_n)$$

However, because this is a missing information problem (the correspondences between observed data and model points are unknown), maximization over this function is not reasonable. EM methods can cope with this difficulty by assigning the missing data to a set of *hidden random* *variables*. In this case, the authors use the notation $\mathcal{Z} = \{Z_j\}_{1\le j\le m}$ to represent these hidden variables. These were seen above, and it was stated that Z_j assigns the data point $\boldsymbol{Y}_j$ to the model point $\boldsymbol{X}_i$ (or to the outlier cluster). The authors use a suggestion from Dempster et al. to replace $\mathcal{L}$ with the *expected complete-data log-likelihood conditioned by the observed data*.

To denote the expected complete-data log-likelihood conditioned by the observed data, one writes

$$\mathcal{E}(\boldsymbol{\Psi}|\mathcal{Y},\mathcal{Z}) = E_Z[\log P(\mathcal{Y},\mathcal{Z};\boldsymbol{\Theta},\boldsymbol{\Sigma_1},\dots,\boldsymbol{\Sigma}_n) \mid \mathcal{Y}]$$

Note the appearance of both the $E_Z[\cdot]$ function and the condition on $\mathcal{Y}$, as well as the inclusion of $\mathcal{Z}$ (which indicates that the data is complete, consisting of both observed and missing information). The expectation is taken over $\mathcal{Z}$ because it is the only random part of the data. Using the notation $\boldsymbol{\Psi} = \{\boldsymbol{\Theta},\boldsymbol{\Sigma_1},\dots,\boldsymbol{\Sigma}_n\}$, the authors rewrite this expression as

$$\mathcal{E}(\boldsymbol{\Psi}|\mathcal{Y},\mathcal{Z}) = \sum_{\mathcal{Z}} P(\mathcal{Z}|\mathcal{Y},\boldsymbol{\Psi}) \log P(\mathcal{Y},\mathcal{Z};\boldsymbol{\Psi})$$

This leap is often confusing. In fact, $\mathcal{E}(\boldsymbol{\Psi}|\mathcal{Y},\mathcal{Z})$ is also known as the Q-function, and a joke among researchers says that it is short for "quixotic function." The authors do not state the steps required to rewrite the equation in this way, but they do cite *Pattern and Machine Learning* by Bishop. The missing step is found on page 20 of the text:

$$\mathcal{E}(\boldsymbol{\Psi}|\mathcal{Y},\mathcal{Z}) \sim E_x(f|y) = \sum_x p(x|y) f(x)$$

CONDITIONAL EXPECTATION

In this case, $f = \log P(\mathcal{Y},\mathcal{Z};\boldsymbol{\Psi})$, $y = \mathcal{Y}$, and $x = \mathcal{Z}$. Making these substitutions gives

$$E_Z[\log P(\mathcal{Y},\mathcal{Z};\boldsymbol{\Psi}) \mid \mathcal{Y}] = \sum_{\mathcal{Z}} p(\mathcal{Z}|\mathcal{Y}) \log P(\mathcal{Y},\mathcal{Z};\boldsymbol{\Psi})$$

The authors add $\boldsymbol{\Psi}$ to the $p(\mathcal{Z}|\mathcal{Y})$ term conditional, despite the fact that the formula referenced does not explicitly require this. Their motives are uncertain as this particular formula for $\mathcal{E}(\boldsymbol{\Psi}|\mathcal{Y},\mathcal{Z})$ is *never meaningfully referenced* in the article again. The mysterious inclusion of this expression without explanation is one critique that can be made of the work, especially because confused readers may waste time looking up the citation, only to find that the information contained therein does not exactly match the result printed by the authors.

To continue, an explicit formula for $\mathcal{E}(\boldsymbol{\Psi}|\mathcal{Y},\mathcal{Z})$ is then derived. The authors begin by expanding the logarithmic component, $\log P(\mathcal{Y},\mathcal{Z};\boldsymbol{\Psi})$, as follows:

$$\log P(\mathcal{Y},\mathcal{Z};\boldsymbol{\Psi}) = \log \prod_{j=1}^{m} P(\boldsymbol{Y}_j, Z_j;\boldsymbol{\Psi})$$

The above step has made use of the fact that the log of a product is equal to the sum of the multiplicands. The next step separates $P(\boldsymbol{Y}_j, Z_j;\boldsymbol{\Psi})$ into two terms:

$$\log P(\mathcal{Y},\mathcal{Z};\mathbf{\Psi}) = \log\prod_{j=1}^{m} P\big(\mathbf{Y}_j|Z_j;\mathbf{\Psi}\big)\,P\big(Z_j\big)$$

The above step utilizes the formula *total probability*, more generally written as

$$P(\mathcal{Y},\mathcal{Z};\mathbf{\Psi}) \sim P(A) = \sum_{n} P(A|B_n)P(B_n)$$

Next, the authors write

$$\log P(\mathcal{Y},\mathcal{Z};\mathbf{\Psi}) = \log\prod_{j=1}^{m}\prod_{i=1}^{n+1}\big\{p_i P\big(\mathbf{Y}_j|Z_j = i;\mathbf{\Psi}\big)\big\}^{\delta_{iZ_j}}$$

where δ_{iZ_j} is the Kronecker symbol defined by

$$\delta_{iZ_j} = \begin{cases} 1, & \text{if } Z_j = i \\ 0, & \text{otherwise} \end{cases}$$

The main idea here is that the term $p_i P\big(\mathbf{Y}_j|Z_j = i;\mathbf{\Psi}\big)$ will be taken to the 0^{th} power (and thus transformed into 1) if the assignment made by the missing data is an <u>incorrect</u> one, and it will contribute to the log-likelihood otherwise. The product operators indicate that all m observed points will contribute to the value of $\log P(\mathcal{Y},\mathcal{Z};\mathbf{\Psi})$ and also that all $n+1$ possible match assignments will be considered (with only the correct ones contributing to $\log P(\mathcal{Y},\mathcal{Z};\mathbf{\Psi})$). Furthermore, the appearance of p_i and $Z_j = i$ is precipitated by the second product operator. This explains all the steps taken by the authors to expand $\mathcal{E}(\mathbf{\Psi}|\mathcal{Y},\mathcal{Z})$.

For the "grand finale," the authors substitute the expansion of $\log P(\mathcal{Y},\mathcal{Z};\mathbf{\Psi})$ back into $\mathcal{E}(\mathbf{\Psi}|\mathcal{Y},\mathcal{Z})$ to obtain an explicit formula useful for estimating parameters:

$$\mathcal{E}(\mathbf{\Psi}|\mathcal{Y},\mathcal{Z}) = E_Z[\log P(\mathcal{Y},\mathcal{Z};\mathbf{\Psi}) \mid \mathcal{Y}]$$

becomes

$$\mathcal{E}(\mathbf{\Psi}|\mathcal{Y},\mathcal{Z}) = E_Z\left[\log\prod_{j=1}^{m}\prod_{i=1}^{n+1}\big\{p_i P\big(\mathbf{Y}_j|Z_j = i;\mathbf{\Psi}\big)\big\}^{\delta_{iZ_j}} \mid \mathcal{Y}\right]$$

The authors do not show the step in which the product operators are eliminated and replaced with summation operators. The full work is given below. The first step is to replace the operators:

$$\mathcal{E}(\mathbf{\Psi}|\mathcal{Y},\mathcal{Z}) = E_Z\left[\sum_{j=1}^{m}\sum_{i=1}^{n+1}\delta_{iZ_j}\big\{\log p_i + \log P\big(\mathbf{Y}_j|Z_j = i;\mathbf{\Psi}\big)\big\} \mid \mathcal{Y}\right]$$

Note that the Kronecker symbol has been pulled out of the exponent due using the properties of the log function. Because only the Kronecker symbol explicitly depends on Z, the expression for $\mathcal{E}(\mathbf{\Psi}|\mathcal{Y},\mathcal{Z})$ can be rewritten as

$$\sum_{j=1}^{m}\sum_{i=1}^{n+1} E_Z\left[\delta_{iZ_j}|\mathcal{Y}\right]\{\log p_i + \log P(\boldsymbol{Y}_j|Z_j = i;\boldsymbol{\Psi})\}$$

The authors do not describe how to determine the value of $E_Z\left[\delta_{iZ_j}|\mathcal{Y}\right]$. The missing step is filled in by returning to the *conditional expectation equation* above:

$$E_Z\left[\delta_{iZ_j}|\mathcal{Y}\right] \sim E_x(f|y) = \sum_x p(x|y)f(x)$$

CONDITIONAL EXPECTATION

Here, $f = E_Z\left[\delta_{iZ_j}|\mathcal{Y}\right]$, $y = \mathcal{Y}$, and $x = \mathcal{Z}$. Therefore,

$$E_Z\left[\delta_{iZ_j}|\mathcal{Y}\right] = \sum_Z \delta_{iZ_j} p(\mathcal{Z}|\mathcal{Y})$$

Recall that

$$\mathcal{Z} = \{\boldsymbol{Z}_j\}_{1\le j\le m}$$

Where $\boldsymbol{Z}_j$ can take on values from $1 \ldots n+1$, assigning correspondences for each of the m observed data points $\boldsymbol{Y}_j$ to a model point $\boldsymbol{X}_i$. If the summation operator is made to sum from $k = 1 \ldots n+1$, the expression for $E_Z\left[\delta_{iZ_j}|\mathcal{Y}\right]$ becomes

$$E_Z\left[\delta_{iZ_j}|\mathcal{Y}\right] = \sum_{k=1}^{n+1} \delta_{ik} P\left(Z_j = k\middle|\boldsymbol{Y}_j\right)$$

Here, the Kronecker is written simply as δ_{ik} because it appears alongside $P\left(Z_j = k\middle|\boldsymbol{Y}_j\right)$. The symbol $\mathcal{Y}$ has been replaced with $\boldsymbol{Y}_j$ because $\mathcal{Z}$ has been rewritten as $Z_j = k$, which cannot be conditioned on the entirety of $\mathcal{Y}$, but only on the observed data point $\boldsymbol{Y}_j$ to which it is linked. Note that $E_Z\left[\delta_{iZ_j}|\mathcal{Y}\right]$ is therefore equal to the posterior, or α_{ji}.

$$E_Z\left[\delta_{iZ_j}|\mathcal{Y}\right] = \alpha_{ji}$$

To obtain the key parameter estimation equation of the complete-data log-likelihood, three more manipulations are necessary. Substituting the result above into $\mathcal{E}(\boldsymbol{\Psi}|\mathcal{Y},\mathcal{Z})$ gives

$$\mathcal{E}(\boldsymbol{\Psi}|\mathcal{Y},\mathcal{Z}) = \sum_{j=1}^{m}\sum_{i=1}^{n+1} \alpha_{ji}\{\log p_i + \log P(\boldsymbol{Y}_j|Z_j = i;\boldsymbol{\Psi})\}$$

Now, the authors substitute the *Gaussian* and *uniform* distributions from the derivation of α_{ji} into the expression. They do not show any steps, but all the necessary work is displayed here.

$$\mathcal{E}(\boldsymbol{\Psi}) = \sum_{j=1}^{m}\sum_{i=1}^{n} \alpha_{ji}\{\log p_i + \log \mathcal{N}[\boldsymbol{Y}_j|\,\boldsymbol{\mu}(\boldsymbol{X}_i;\boldsymbol{\Theta}),\boldsymbol{\Sigma}_i]\}$$

Note that the upper index of the second summation operator has decreased by one because there are only n model points. Also note that $\mathcal{E}(\boldsymbol{\Psi}|\mathcal{Y},\mathcal{Z})$ has become $\mathcal{E}(\boldsymbol{\Psi})$ because the expression is now explicit. Substituting for the fully parameterized expression of $\mathcal{N}\left[\boldsymbol{Y}_j|\ \boldsymbol{\mu}(\boldsymbol{X}_i;\boldsymbol{\Theta}),\boldsymbol{\Sigma}_i\right]$ gives

$$\mathcal{E}(\boldsymbol{\Psi}) = \sum_{j=1}^{m}\sum_{i=1}^{n}\alpha_{ji}\Big\{\log p_i + \log\Big\{\left[(2\pi)^{3/2}|\boldsymbol{\Sigma}_i|^{1/2}\right]^{-1}\exp\left[-\frac{1}{2}\left(\boldsymbol{Y}_j-\boldsymbol{\mu}(\boldsymbol{X}_i;\boldsymbol{\Theta})\right)^{\mathrm{T}}\boldsymbol{\Sigma}_i^{-1}\left(\boldsymbol{Y}_j-\boldsymbol{\mu}(\boldsymbol{X}_i;\boldsymbol{\Theta})\right)\right]\Big\}\Big\}$$

Substituting for the Mahalanobis distance (as before) and eliminating constants gives

$$\mathcal{E}(\boldsymbol{\Psi}) = \sum_{j=1}^{m}\sum_{i=1}^{n}\alpha_{ji}\left\{\log p_i + \log\left[|\boldsymbol{\Sigma}_i|^{-1/2}\exp\left(-\frac{1}{2}\left\|\boldsymbol{Y}_j-\boldsymbol{\mu}(\boldsymbol{X}_i;\boldsymbol{\Theta})\right\|_{\Sigma_i}^2\right)\right]\right\}$$

Recall the parameterization of p_i, the priors:

$$p_i = \text{priors} = P(Z_j = i) = \begin{cases} p_{in} = \dfrac{v}{V}, & \text{if } 1 \le i \le n \\ p_{out} = \dfrac{V-nv}{V}, & \text{if } i = n+1 \end{cases}$$

PARAMETERIZATION OF PRIORS

The values of p_i are in no way dependent on $\boldsymbol{\Psi}$, and p_i may therefore be eliminated from the expression for $\mathcal{E}(\boldsymbol{\Psi})$ without affecting the outcome of the ECM procedure, leaving

$$\mathcal{E}(\boldsymbol{\Psi}) = \sum_{j=1}^{m}\sum_{i=1}^{n}\alpha_{ji}\left\{\log\left[|\boldsymbol{\Sigma}_i|^{-1/2}\exp\left(-\frac{1}{2}\left\|\boldsymbol{Y}_j-\boldsymbol{\mu}(\boldsymbol{X}_i;\boldsymbol{\Theta})\right\|_{\Sigma_i}^2\right)\right]\right\}$$

Applying the properties of the log function to simplify this expression gives the result of the authors:

$$\boxed{\mathcal{E}(\boldsymbol{\Psi}) = -\frac{1}{2}\sum_{j=1}^{m}\sum_{i=1}^{n}\alpha_{ji}\left(\log|\boldsymbol{\Sigma}_i| + \left\|\boldsymbol{Y}_j-\boldsymbol{\mu}(\boldsymbol{X}_i;\boldsymbol{\Theta})\right\|_{\Sigma_i}^2\right)}$$

Horaud et al., Equation (18), page 591.

The maximization of $\mathcal{E}(\boldsymbol{\Psi})$ will also maximize the observed-data log-likelihood, and the parameters which maximize that log-likelihood are desired. In other words, if $\boldsymbol{\Psi}^q = \left\{\boldsymbol{\Theta}^q,\boldsymbol{\Sigma}_1^q,\dots,\boldsymbol{\Sigma}_n^q\right\}$ is the current best estimate of the optimal parameters, then

$$\boldsymbol{\Psi}^{q+1} = \arg\max_{\boldsymbol{\Psi}}\left[-\frac{1}{2}\sum_{j=1}^{m}\sum_{i=1}^{n}\alpha_{ji}^q\left(\log|\boldsymbol{\Sigma}_i^q| + \left\|\boldsymbol{Y}_j-\boldsymbol{\mu}(\boldsymbol{X}_i;\boldsymbol{\Theta})\right\|_{\Sigma_i^q}^2\right)\right]$$

THEORETICAL OBJECTIVE FUNCTION

However, the authors note that in practice, it is better first to maximize the objective function over $\boldsymbol{\Theta}$ only while holding the covariance matrices $\boldsymbol{\Sigma}_1,\dots,\boldsymbol{\Sigma}_n$ constant (lest a difficult nonlinear maximization problem arise) and *then* to estimate the new covariance matrices using the new registration parameter values $\boldsymbol{\Theta}^*$. Therefore, the maximization step is replaced by two conditional maximization steps: one for the registration parameters $\mathbf{\Theta}$, and one for the covariance matrices $\mathbf{\Sigma}_1, \dots, \mathbf{\Sigma}_n$.

The authors elect to do away with the negative sign in $\mathbf{\Psi}^{q+1}$ prior to splitting it into two steps. This means the steps will instead be *conditional minimizations*:

$$\mathbf{\Theta}^{q+1} = \arg\min_{\mathbf{\Theta}} \frac{1}{2} \sum_{j=1}^{m} \sum_{i=1}^{n} \alpha_{ji}^{q} \left\| \boldsymbol{Y}_j - \boldsymbol{\mu}(\boldsymbol{X}_i; \mathbf{\Theta}) \right\|_{\Sigma_i^q}^2$$

CONDITIONAL MINIMIZATION TO OBTAIN REG. PARAMS.

Horaud et al., Equation (19), page 591.

The authors assume that the reader knows how to obtain the covariance matrices and cite no source for the procedure leading to Equation (20) in the article. A helpful guide to the EM algorithm by Chen and Gupta, 2010 provides the steps. In the guide, the math proceeds as follows (caution: the notation and equations are not exactly the same, but they are analogous):

Define the Q-function as

$$Q(\theta|\theta^{(m)}) = \sum_{i=1}^{n} \sum_{j=1}^{k} \gamma_{ij}^{(m)} \left[\log w_j - \frac{1}{2} \log|\mathbf{\Sigma}_j| - \frac{1}{2} (y_i - \mu_j)^{\mathrm{T}} \mathbf{\Sigma}_j^{-1} (y_i - \mu_j) \right]$$

Make the substitution

$$n_j^{(m)} = \sum_{i=1}^{n} \gamma_{ij}^{(m)}$$

and take the partial derivative with respect to $\mathbf{\Sigma}_j$ to receive

$$\frac{\partial Q(\theta|\theta^{(m)})}{\partial \mathbf{\Sigma}_j} = -\frac{1}{2} n_j^{(m)} \frac{\partial}{\partial \mathbf{\Sigma}_j} \log|\mathbf{\Sigma}_j| - \frac{1}{2} \sum_{i=1}^{n} \gamma_{ij}^{(m)} \frac{\partial}{\partial \mathbf{\Sigma}_j} (y_i - \mu_j)^{\mathrm{T}} \mathbf{\Sigma}_j^{-1} (y_i - \mu_j)$$

$$\frac{\partial Q(\theta|\theta^{(m)})}{\partial \mathbf{\Sigma}_j} = -\frac{1}{2} n_j^{(m)} \mathbf{\Sigma}_j^{-1} + \frac{1}{2} \sum_{i=1}^{n} \gamma_{ij}^{(m)} \mathbf{\Sigma}_j^{-1} (y_i - \mu_j)(y_i - \mu_j)^{\mathrm{T}} \mathbf{\Sigma}_j^{-1}$$

Set this derivative to zero:

$$\frac{\partial Q(\theta|\theta^{(m)})}{\partial \mathbf{\Sigma}_j} = 0, \qquad j = 1, \dots, k$$

And solve it for $\mathbf{\Sigma}_j$ to obtain an expression for $\mathbf{\Sigma}_j^*$.

An analogous procedure for the case in the article involves setting the partial derivative of $\mathcal{E}(\mathbf{\Psi})$ with respect to $\mathbf{\Sigma}_j$ equal to zero:

$$\frac{\partial \mathcal{E}(\mathbf{\Psi})}{\partial \mathbf{\Sigma}_j} = 0$$

Solving this for $\mathbf{\Sigma}_j$ provides the formula for $\mathbf{\Sigma}_i^{q+1}$.

$$\mathbf{\Sigma}_i^{q+1} = \frac{\sum_{j=1}^{m} \alpha_{ji}^{q} \left(\boldsymbol{Y}_j - \boldsymbol{\mu}(\boldsymbol{X}_i; \mathbf{\Theta}^{q+1})\right)\left(\boldsymbol{Y}_j - \boldsymbol{\mu}(\boldsymbol{X}_i; \mathbf{\Theta}^{q+1})\right)^{\mathrm{T}}}{\sum_{j=1}^{m} \alpha_{ji}^{q}}$$

EXPRESSION TO OBTAIN COVARIANCES

Horaud et al., Equation (20), page 592

The authors turn their attention once again to the formula for $\mathbf{\Theta}^{q+1}$ to simplify it. They define quantities known as the *virtual observations* $\boldsymbol{W}_i$ for the points assigned to model points $\boldsymbol{X}_i$.

$$\boldsymbol{W}_i = \frac{1}{\lambda_i} \sum_{j=1}^{m} \alpha_{ji} \boldsymbol{Y}_j$$

VIRTUAL OBSERVATIONS

Horaud et al., Equation (22), page 592

The weight of virtual observation i is given by λ_i.

$$\lambda_i = \sum_{j=1}^{m} \alpha_{ji}$$

VIRTUAL WEIGHTS

Horaud et al., Equation (23), page 592

The motivation for this substitution is purely simplification. As such, no explanation for the variable assignments is given; substituting these quantities into the formula for $\mathbf{\Theta}^{q+1}$ gives

$$\mathbf{\Theta}^{q+1} = \arg\min_{\mathbf{\Theta}} \frac{1}{2} \sum_{i=1}^{n} \lambda_i^{q} \left\| \boldsymbol{W}_i^{q} - \boldsymbol{\mu}(\boldsymbol{X}_i; \mathbf{\Theta}) \right\|_{\Sigma_i^q}^{2}$$

CONDITIONAL MINIMIZATION TO OBTAIN REG. PARAMS.

Horaud et al., Equation (24), page 592

Now that a computationally efficient formula for $\mathbf{\Theta}^{q+1}$ has been obtained, the formula for $\boldsymbol{t}^*$ may be derived. Recall that for a rigid transformation,

$$\boldsymbol{\mu}(\boldsymbol{X}_i; \mathbf{\Theta}) = \underset{3\text{x}3}{\mathbf{R}} \boldsymbol{X}_i + \underset{3\text{x}1}{\boldsymbol{t}}, \qquad \mathbf{\Theta} := \left\{ \underset{3\text{x}3}{\mathbf{R}}, \underset{3\text{x}1}{\boldsymbol{t}} \right\}$$

The authors substitute for $\boldsymbol{\mu}(\boldsymbol{X}_i; \mathbf{\Theta})$ in the formula for $\mathbf{\Theta}^{q+1}$, giving

$$\boldsymbol{\Theta}^* = \arg\min_{\mathbf{R},\boldsymbol{t}} \frac{1}{2}\sum_{i=1}^{n} \lambda_i^q \left\|\boldsymbol{W}_i^q - \mathbf{R}\boldsymbol{X}_i + \boldsymbol{t}\right\|_{\Sigma_i^q}^2$$

CONDITIONAL MINIMIZATION TO OBTAIN REG. PARAMS.

Horaud et al., Equation (26), page 592

The authors next derive an expression for $\boldsymbol{t}^*$, despite the fact that the algorithm calls for the computation of $\mathbf{R}^*$ first. This is considered to be a confusing notational error on the part of the authors. In any case, $\boldsymbol{t}^*$ is obtained by setting the derivative of $\boldsymbol{\Theta}^*$ with respect to $\boldsymbol{t}$ to zero and solving for $\boldsymbol{t}$. That is, the equation

$$\frac{\partial \boldsymbol{\Theta}^*}{\partial \boldsymbol{t}} = 0$$

is solved for $\boldsymbol{t}$ to obtain $\boldsymbol{t}^*$. The result is

$$\boldsymbol{t}^* = \left(\sum_{i=1}^{n} \lambda_i \boldsymbol{\Sigma}_i^{-1}\right)^{-1} \sum_{i=1}^{n} \lambda_i \boldsymbol{\Sigma}_i^{-1}(\boldsymbol{W}_i - \mathbf{R}\boldsymbol{X}_i)$$

UPDATED TRANSLATION COMPONENT

Horaud et al., Equation (27), page 593

The authors proceed to substitute the formula for $\boldsymbol{t}^*$ into the formula for $\boldsymbol{\Theta}^*$. A large amount of algebra is required to expand the expression, which the authors have kindly performed.

$$\mathbf{R}^* = \arg\min_{\mathbf{R}} \frac{1}{2}\sum_{i=1}^{n} \lambda_i\left(\boldsymbol{X}_i^{\mathrm{T}}\mathbf{R}^{\mathrm{T}}\boldsymbol{\Sigma}_i^{-1}\mathbf{R}\boldsymbol{X}_i + 2\boldsymbol{X}_i^{\mathrm{T}}\mathbf{R}^{\mathrm{T}}\boldsymbol{\Sigma}_i^{-1}\boldsymbol{t}^* - 2\boldsymbol{X}_i^{\mathrm{T}}\mathbf{R}^{\mathrm{T}}\boldsymbol{\Sigma}_i^{-1}\boldsymbol{W}_i - 2\boldsymbol{t}^{*\mathrm{T}}\boldsymbol{\Sigma}_i^{-1}\boldsymbol{W}_i + \boldsymbol{t}^{*\mathrm{T}}\boldsymbol{\Sigma}_i^{-1}\boldsymbol{t}^*\right)$$

UPDATED ROTATION COMPONENT

Horaud et al., Equation (28), page 593

The authors proceed to demonstrate that the formulas for both $\mathbf{R}^*$ and $\boldsymbol{t}^*$ may be conveniently simplified in the case of *isotropic covariances*. Isotropic covariance matrices have the form $\boldsymbol{\Sigma}_i = \sigma_i^2\mathbf{I}_3$, where σ_i^2 is the variance of the Gaussian indexed by i. The authors point out that in this case, the Mahalanobis distance will reduce to the Euclidean distance, simplifying some expressions. This is an important point because according to the authors, many PR methods utilize isotropic covariances.

On the other hand, the authors go on to advocate for the use of *anisotropic covariances.* They formulate a solution for $\mathbf{R}^*$ in the anisotropic case by casting it into a *convex optimization criterion*. Their final result is

$$\begin{cases} (\boldsymbol{\rho}^*, \boldsymbol{r}^*) = \arg\min_{(\boldsymbol{\rho},\boldsymbol{r})}\left(\langle \mathbf{A}, \boldsymbol{\rho}\rangle + 2\boldsymbol{b}^{\mathrm{T}}\boldsymbol{r}\right), \\ \langle \boldsymbol{\Delta}_{kl}, \boldsymbol{\rho}\rangle = \delta_{kl}, \quad k = 1,2,3;\ l = 1,2,3, \\ \boldsymbol{\rho} \succcurlyeq \boldsymbol{r}\boldsymbol{r}^{\mathrm{T}}. \end{cases}$$

ANISOTROPIC COVARIANCE MODEL

Horaud et al., Equation (34), page 593

The authors define the variables as in Table 1. However, their explanation of the variable $\boldsymbol{\Delta}_{kl}$ is insufficient. They state that "the entries of the six 9×9 matrices $\boldsymbol{\Delta}_{kl}$ are easily obtained from the constraint $\mathbf{R}\mathbf{R}^{\mathrm{T}} = \mathbf{I}$." It is likely that $\boldsymbol{\Delta}_{kl}$ is *tenor notation* for the *mixed rank-2 tensor* version of Kronecker's symbol, δ_{kl}. In this case, the dot product $\langle \boldsymbol{\Delta}_{kl}, \boldsymbol{\rho}\rangle$ would yield δ_{kl} because $\langle \boldsymbol{\Delta}_{kl}, \boldsymbol{\rho}\rangle = \boldsymbol{\Delta}_{kl} \cdot \boldsymbol{\rho} = \mathrm{trace}\left(\boldsymbol{\Delta}_{kl}\boldsymbol{\rho}^{\mathrm{T}}\right)$. Since $\boldsymbol{\rho}$ is rank-one, then the trace of the 9×9 matrix $\boldsymbol{\Delta}_{kl}\boldsymbol{\rho}^{\mathrm{T}}$ will be equal to 1 or 0.

The anisotropic covariance model requires the matrix $\boldsymbol{\rho}$ to be positive semidefinite, but the authors do not explain what this means. The definition of a positive semidefinite matrix is provided here.

The 3×3 matrix $\mathbf{M}$ is *positive semidefinite* if for any 3×1 vector $\boldsymbol{x}$,

$$\boldsymbol{x}^{\mathrm{T}}\mathbf{M}\boldsymbol{x} \geq 0$$

This completes the description of the anisotropic covariance model proposed by the authors. They now extend the ECMPR-Rigid algorithm to the problem of articulated point registration.

Table 1 – Quadratic framework variables.

Quantity	Definition	Comment
$\boldsymbol{r}$	$\mathrm{vec}(\mathbf{R})$	The 9×2 vector containing the entries of $\mathbf{R}$.
$\boldsymbol{\rho}$	$\boldsymbol{r}\boldsymbol{r}^{\mathrm{T}}$	Rank-one positive-symmetric matrix derived from $\mathbf{R}$.
$\boldsymbol{\Delta}_{kl}$	**undefined**	n/a
$\mathrm{vec}(\mathbf{A})$	$(A_{11} \ldots A_{mn})^{\mathrm{T}}$	Vectorization function.
$\mathbf{A}$	$\mathbf{N} - \mathbf{M}^{\mathrm{T}}\mathbf{K}\mathbf{M}$	9×9 matrix derived from the expansion of Equation (28).
$\boldsymbol{b}$	$\mathbf{M}^{\mathrm{T}}\boldsymbol{p} - \boldsymbol{q}$	9×1 vector used in the rearrangement of Equation (28).
$\mathbf{N}$	$\sum_{i=1}^{n} \lambda_i \boldsymbol{X}_i \boldsymbol{X}_i^{\mathrm{T}} \otimes \boldsymbol{\Sigma}_i^{-1}$	9×9 matrix used in the rearrangement of Equation (28).
$\mathbf{M}$	$\sum_{i=1}^{n} \lambda_i \boldsymbol{X}_j^{\mathrm{T}} \otimes \boldsymbol{\Sigma}_i^{-1}$	3×9 matrix used in the rearrangement of Equation (28).
$\mathbf{K}$	$\left(\sum_{i=1}^{n} \lambda_i \boldsymbol{\Sigma}_i^{-1}\right)^{-1}$	3×3 matrix used in the rearrangement of Equation (28).
$\boldsymbol{p}$	$\mathbf{K}\left(\sum_{i=1}^{n} \lambda \boldsymbol{\Sigma}_i^{-1} \boldsymbol{W}_i\right)$	3×1 vector used in the rearrangement of Equation (28).

$\boldsymbol{q}$	$\operatorname{vec}\left(\sum_{i=1}^{n} \lambda \boldsymbol{\Sigma}_i^{-1} \boldsymbol{W}_i \boldsymbol{X}_i^{\mathrm{T}}\right)$	9×1 vector used in the rearrangement of Equation (28).
$\mathbf{A} \otimes \mathbf{B}$	$\begin{bmatrix} A_{11}\mathbf{B} & \cdots & A_{1n}\mathbf{B} \\ \vdots & \ddots & \vdots \\ A_{m1}\mathbf{B} & \cdots & A_{mn}\mathbf{B} \end{bmatrix}$	Kronecker product.

Articulated Point Registration

The problem of articulated point registration requires estimation of the pose of an articulated shape, such as a human hand. It can be solved by applying the principles behind rigid point registration to the individual rigid segments of the articulated shape (i.e. the palm of the hand, and each component of all five digits).

The article utilizes a modified *open kinematic chain* to describe the hand model. An open kinematic chain consists of a series of movements of a linked segment (in this case, a jointed finger) whose distal end is free in space (the fingertip has no distal attachment, of course). The authors consider the root of the chain (in this case, the palm) to possess six degrees of freedom (three rotations and three translations), while the segments (the fingers) possess between one and three.

In this model, the root has *free* motion while the fingers have *constrained* motion. The relevant transformation in this case is

$$\boldsymbol{\mu}\left(\boldsymbol{X}_i^{(p)}; \boldsymbol{\Theta}\right) = \mathbf{R}(\boldsymbol{\Theta})\boldsymbol{X}_i^{(p)} + \boldsymbol{t}(\boldsymbol{\Theta}), \qquad \boldsymbol{\Theta} \coloneqq \{\boldsymbol{\Theta}_0, \dots, \boldsymbol{\Theta}_p\}$$

The authors note that the key difference between the articulated point registration transformation and the rigid point registration transformation lies in the kinematic parameters. The rotations and translations were the free parameters in the rigid case, but here, they are constrained by the kinematic parameters, denoted $\boldsymbol{\Theta} \coloneqq \{\boldsymbol{\Theta}_0, \dots, \boldsymbol{\Theta}_p\}$, where $\boldsymbol{\Theta}_0$ represents the motion of the palm and the other $\boldsymbol{\Theta}_1, \dots, \boldsymbol{\Theta}_p$ represent the motions of the other rigid parts (the fingers).

For simplicity, the authors utilize a 4×4 displacement matrix which incorporates both the rotation and translation components of the transformation in a single matrix, $\mathbf{T}_p(\boldsymbol{\Theta})$. They go on to express $\mathbf{T}_p(\boldsymbol{\Theta})$ as a chain of homogeneous transformations:

$$\mathbf{T}_p(\boldsymbol{\Theta}) = \mathbf{Q}_0(\boldsymbol{\Theta}_0)\mathbf{Q}_1(\boldsymbol{\Theta}_1) \dots \mathbf{Q}_p(\boldsymbol{\Theta}_p)$$

Where $\mathbf{Q}_0$ describes the free motion of the palm, parameterized by $\boldsymbol{\Theta}_0 = \{\operatorname{vec}(\mathbf{R}_0), \boldsymbol{t}_0\}$. Each $\mathbf{Q}_p$ has both a *fixed motion component* (a change of coordinates) and a *constrained motion component* (parameterize by between one and three angles).

Since estimating the entire parameter vector $\boldsymbol{\Theta}$ all at once presents difficulties similar to those seen in the rigid case, a similar solution involving a piece-by-piece estimation strategy is proposed by the authors. One rigid segment is considered at a time instead of simultaneously solving for the parameters of all rigid segments. The authors point out that at least five other approaches have advocated for the latter procedure. To prepare for the derivation of the optimal parameters via conditional minimization expressions, the authors specify the following notations:

$$\mathbf{Q}_0 = \begin{bmatrix} \mathbf{R}_0 & \boldsymbol{t}_0 \\ \mathbf{0} & 1 \end{bmatrix}, \qquad \mathbf{Q}_p = \begin{bmatrix} \mathbf{R}_p & 0 \\ \mathbf{0} & 1 \end{bmatrix}$$

$$\mathbf{T}_p = \mathbf{T}_{p-1}\mathbf{Q}_p = \begin{bmatrix} \mathbf{R}_{0,p-1}\mathbf{R}_p & \boldsymbol{t}_{0,p-1} \\ \mathbf{0}^{\mathrm{T}} & 1 \end{bmatrix}$$

$$\boldsymbol{\Theta}_0^* = \arg\min_{\mathbf{R}_0,\boldsymbol{t}_0} \frac{1}{2}\sum_{i=1}^{n_0} \lambda_i \left\| \boldsymbol{W}_i - \mathbf{R}_0\boldsymbol{X}_i^{(0)} + \boldsymbol{t}_0 \right\|_{\Sigma_i}^2$$

CONDITIONAL MINIMIZATION TO OBTAIN ROOT PARAMS.

$$\mathbf{R}_p^* = \arg\min_{\mathbf{R}_p} \frac{1}{2}\sum_{i=1}^{n_p} \lambda_i \left\| \boldsymbol{W}_i - \mathbf{R}_{0,p-1}\boldsymbol{X}_i^{(p)} + \boldsymbol{t}_{0,p-1} \right\|_{\Sigma_i}^2$$

CONDITIONAL MINIMIZATION TO OBTAIN SEGMENT PARAMS.

The authors simplify this notation by introducing the following substitutions:

$$\mathbf{U}_p = \mathbf{R}_{0,p-1}\mathbf{R}_p\mathbf{R}_{0,p-1}^{\mathrm{T}}$$

$$\boldsymbol{V}_i^{(p)} = \mathbf{R}_{0,p-1}\boldsymbol{X}_i^{(p)}$$

and the second conditional minimization is rewritten as

$$\mathbf{U}_p^* = \arg\min_{\mathbf{U}_p} \frac{1}{2}\sum_{i=1}^{n_p} \lambda_i \left\| \boldsymbol{W}_i - \mathbf{U}_p\boldsymbol{V}_i^{(p)} + \boldsymbol{t}_{0,p-1} \right\|_{\Sigma_i}^2$$

CONDITIONAL MINIMIZATION TO OBTAIN SEGMENT PARAMS.

This does not seem to be a particularly useful simplification because the subscript of $\boldsymbol{t}$ continues to possess the $p-1$ index, but this notation will persist for the remainder of the discussion. In order to use the minimization in Equation (44), the transformation $\mathbf{T}_{p-1}$ must be known. Luckily, it is recursively obtainable because $\mathbf{T}_0 = \mathbf{Q}_0 = \begin{bmatrix} \mathbf{R}_0 & \boldsymbol{t}_0 \\ \mathbf{0} & 1 \end{bmatrix}$ (see above).

The ECMPR-Articulated Algorithm

1. *Rigid registration of the root part*: Initialize the current set of data points $\mathcal{Y}^{(0)}$ with the whole data set. Apply the ECMPR-Rigid algorithm to the data set $\mathcal{Y}^{(0)}$ and the set of model points associated with the root part $\mathcal{X}_0$ in order to estimate the pose of the root part. Compute $\mathbf{T}_0 = \mathbf{Q}_0$ using Equation (37). Classify the data points into inliers and outliers. Remove the inliers from $\mathcal{Y}^{(0)}$ to generate a new data set $\mathcal{Y}^{(1)}$.
2. *For each $p = 1 \ldots P$, rigid registration of the p^{th} part*: Apply the ECMPR-Rigid algorithm to the current set of data points $\mathcal{Y}^{(p)}$ and the set $\mathcal{X}_p$. Estimate $\mathbf{R}_p$ from the formula for $\mathbf{U}_p^*$, which is Equation (44). Compute $\mathbf{Q}_p$ using Equation (37) and $\mathbf{T}_p = \mathbf{T}_{p-1}\mathbf{Q}_p$. Classify the data points into inliers and outliers. Remove the inliers from $\mathcal{Y}^{(p)}$ to generate a new data set $\mathcal{Y}^{(p+1)}$.

Now that the key algorithms have been described, the details of the experiments conducted by the authors will be discussed.

Experiments: ECMPR-Rigid

The performance of the ECMPR-Rigid algorithm was compared to that of the Trimmed Iterative Closest Point algorithm (TriICP). TriICP is a robust implementation of the ICP algorithm which utilizes random sampling. Two different experiments were designed.

Experimental Setup 1

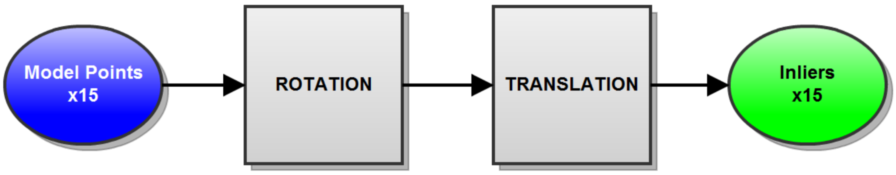

Figure 1 - Uncorrupted inliers.

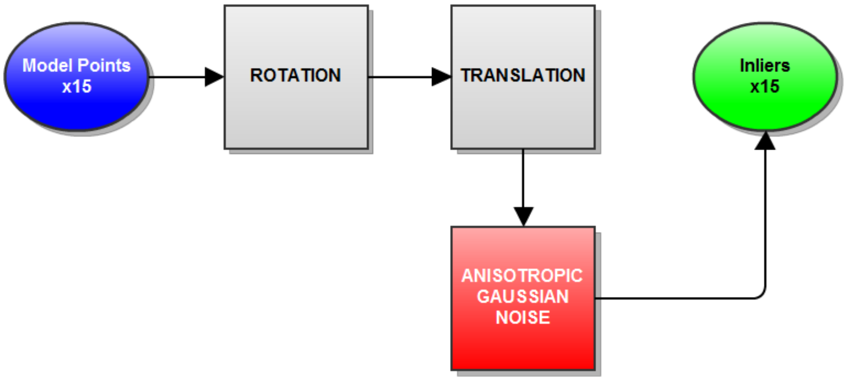

Figure 2 - Corrupted (noisy) inliers.

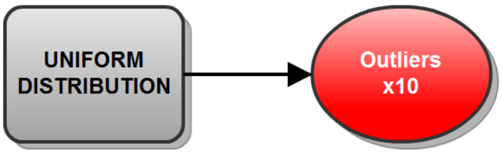

Figure 3 - ECMPR-Rigid Experiment 1 outliers.

Three trials were run for the ECMPR-Rigid algorithm and two trials were run for the TriICP algorithm. The trials varied in their degrees of simulated noise and their types of covariance models. 15 ground truth model points were considered in this experiment. 15 inliers were generated from these model points by rotating them by 25° and randomly translating them. Two trials featured no noise, and both algorithms were 100% effective (although the TriICP algorithm took much longer). The derivation of inliers in this case is described by Figure 1. Figures 2 and 3 diagram the generation of noisy inliers and the outliers. The trials featuring anisotropic Gaussian noise had worse performance than those without noise, and the ECMPR-Rigid algorithm performed better than the TriICP algorithm. The anisotropic Gaussian noise was drawn from two 1D Gaussian probability distributions with two different variances along each dimension, and the variances were allowed to vary between 10 and 100 percent of the box bounding the set of observations. That is, inlier locations were shifted by a value drawn from distributions like those in Figure 4.

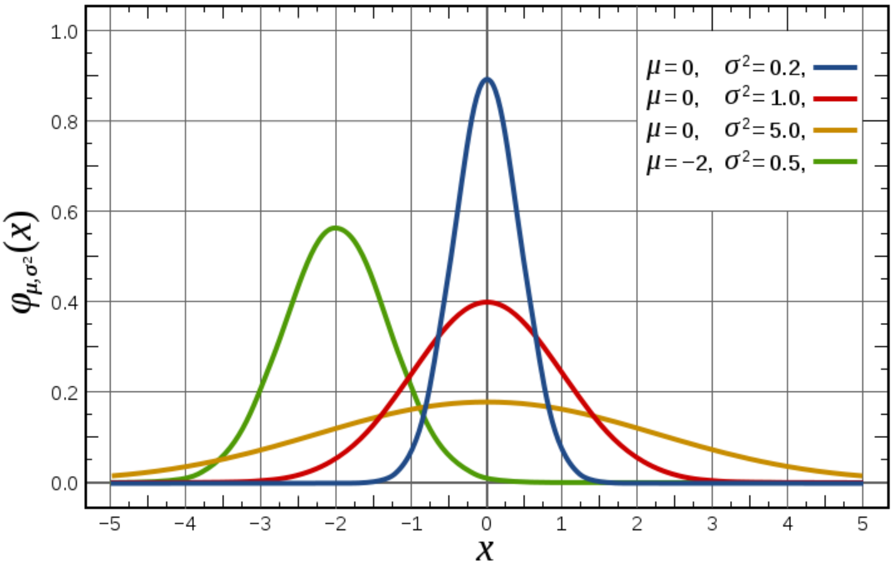

Figure 4 - 1D Gaussian distributions.
http://upload.wikimedia.org/wikipedia/commons/thumb/7/74/Normal_Distribution_PDF.svg/720px-Normal_Distribution_PDF.svg.png

The initialization of parameters for both algorithms was always the same: $\mathbf{R} = \mathbf{I}_3$ and $\boldsymbol{t} = \mathbf{0}$. The authors report that the ECMPR-Rigid algorithm is robust to outliers, which is a valuable property (especially for the articulated point registration case). Since ICP is so sensitive to initialization, it takes many more iterations to finish given such an initialization. Its random sampling strategy for multiple initializations allows it to arrive at a solution, but in much more time than ECMPR-Rigid.

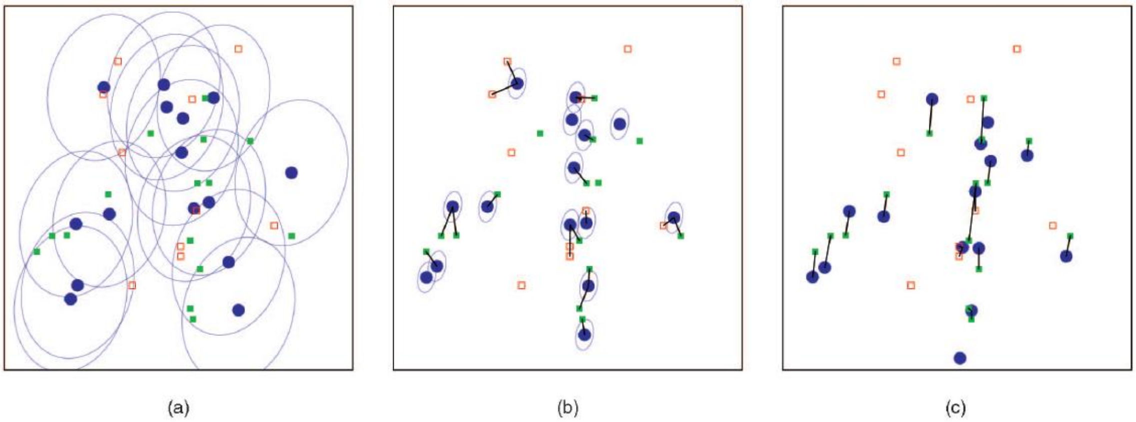

Figure 5 - Iterations of the ECMPR-Rigid algorithm. 2 iterations (a), 6 iterations (b), and 35 iterations. This figure is a reproduction of Horaud et al., Fig. 2.

Figure 5 illustrates the progress of ECMPR-Rigid across a number of iterations. The large blue dots are model points, the green squares are inliers, and the empty red squares are outliers. As the number of iterations increases, the sizes of the covariances decrease.

As an auxiliary result of this experiment, the authors ran 1,000 trials of ECMPR-Rigid in the anisotropic covariance case. These data were treated as a performance measure of the algorithm. The curves plotted by the authors in Figure 4 of their article contain the mean (middle line) and upper and lower standard deviations (bounding lines) of the data from the 1,000 trials. This metric is a useful performance measure because it allows the algorithm's efficacy to be conveyed at a glance.

The authors also report the percentage of correct matches returned by both algorithms. However, they note that the algorithms classify matches differently, and point to the error in rotation and translation as a more reliable measure of comparing the two. In these regards, ECMPR-Rigid outperformed TriICP.

The full results of the experiment are reprinted in Table 1 on page 595 of the original article.

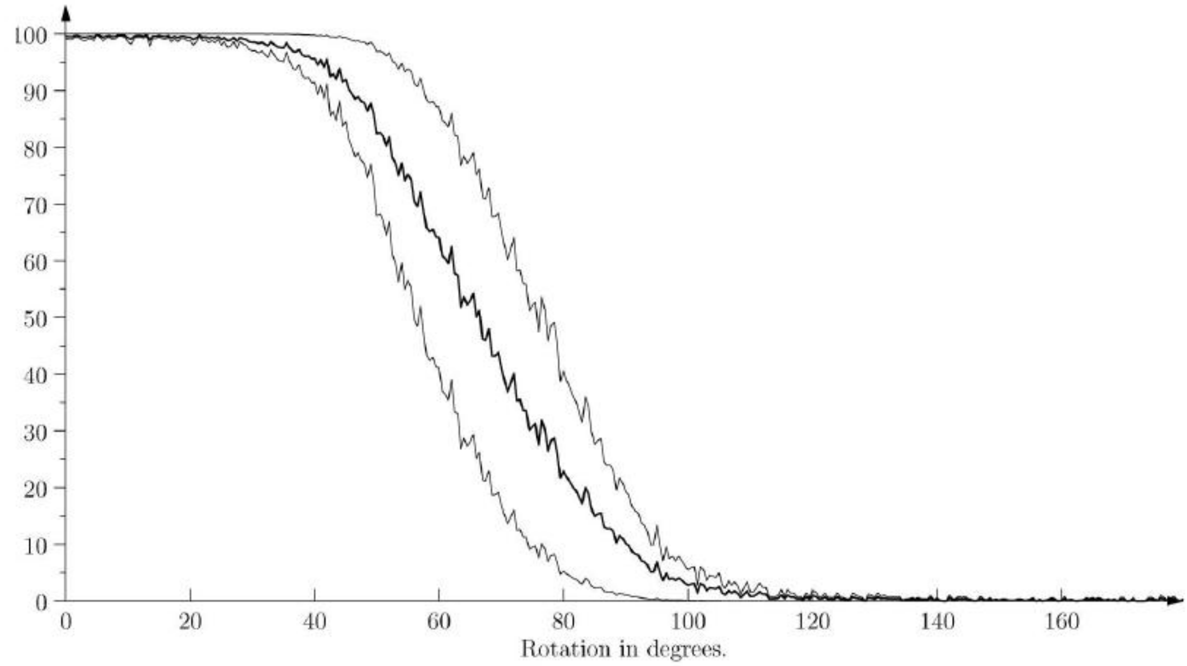

Figure 6 - Correct matches vs. inlier rotation. This figure is reproduction of Hedley et al., Fig. 4a.

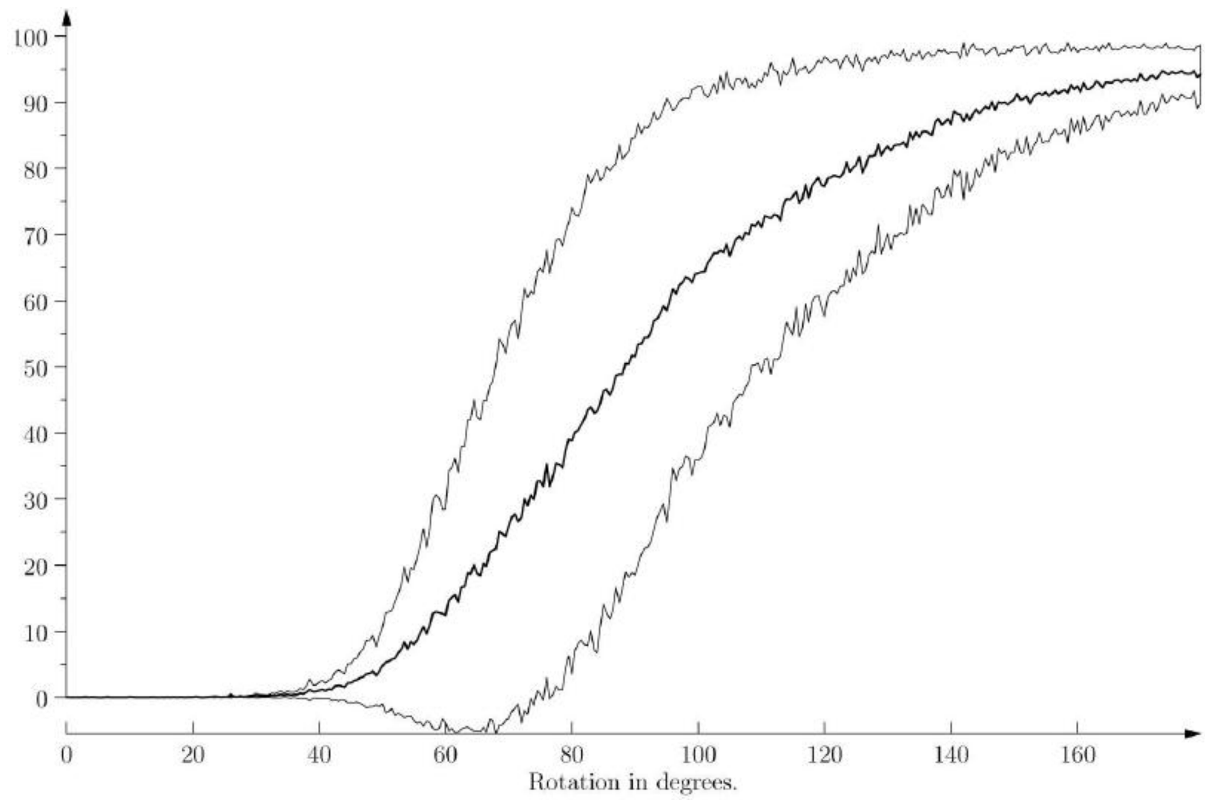

Figure 7 - Error in rotation vs. inlier rotation. This figure is reproduction of Hedley et al., Fig. 4b.

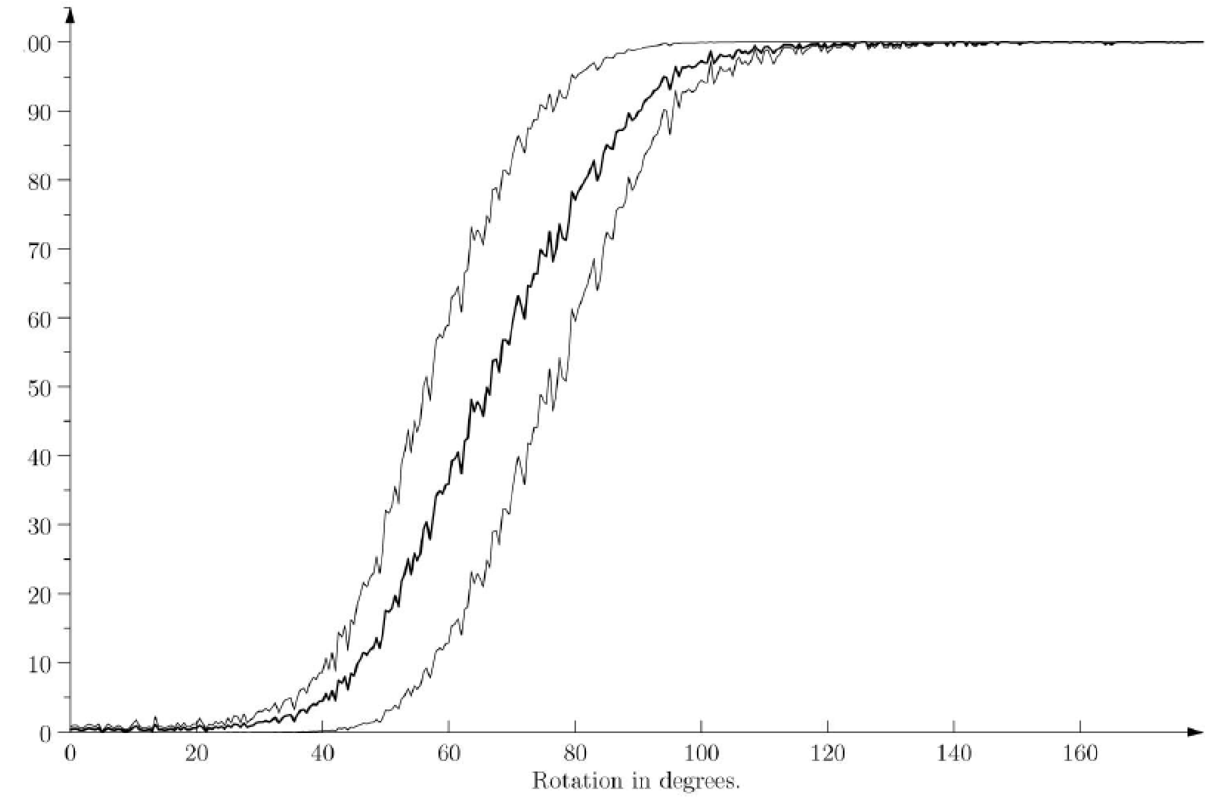

Figure 8 - Error in translation vs. inlier rotation. This figure is reproduction of Hedley et al., Fig. 4c.

Experimental Setup 2

The next ECMPR-Rigid experiment makes use of stereo data. Using a pair of cameras situated at different locations with respect to a walking subject, a pair of images was taken at Time 1 and Time 2, over which the subject translated a distance of 280 mm. Using stereo vision software, a set of 3D points was reconstructed for each image pair. The set of 3D points at Time 1 constituted the "model" points, or the ground truth. The set of 3D points at Time 2 constituted the "observed data" points.

When the two point sets were fed to ECMPR-Rigid, it correctly estimated both the rotation and translation aligning the two image pairs. However, TriICP was only able to recover the correct rotation. The full results of the experiment are reprinted in Table 2 on page 591 of the original article.

The translation error was computed by $\frac{\|\boldsymbol{t}-\boldsymbol{t}_g\|}{\|\boldsymbol{t}_g\|}$ where $\boldsymbol{t}$ is the estimated translation vector and $\boldsymbol{t}_g$ is the ground truth. Also, the minimization error is given by $\sqrt{\left(n_{in}\sum_{i=1}^{n_{in}}\|\boldsymbol{Y}_i - \mathbf{R}\boldsymbol{X}_i - \boldsymbol{t}\|^2\right)^{-1}}$, where n_{in} is the number of inliers estimated by each algorithm.

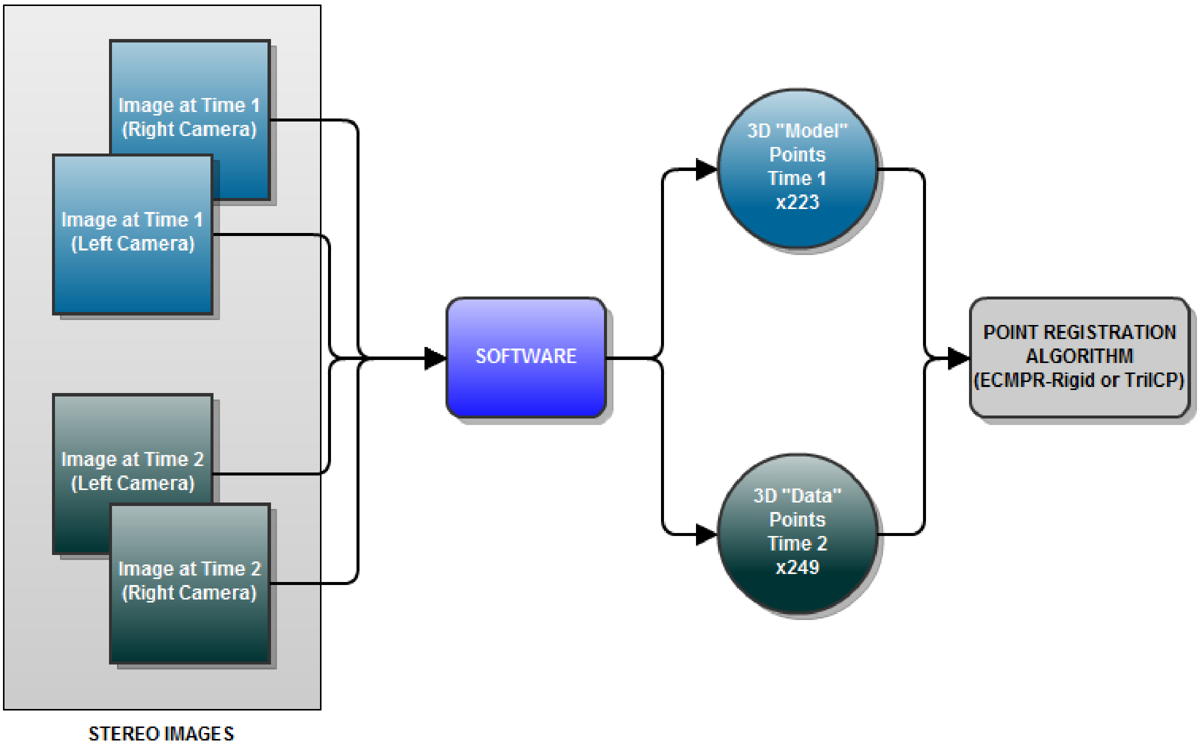

Figure 9 - ECMPR-Rigid Experiment 2 setup summary.

Experiments: ECMPR-Articulated

The performance of the ECMPR-Articulated algorithm was evaluated. Two different experiments were designed.

Experimental Setup 1

The first ECMPR-Articulated experiment was performed on simulated data. Recall that the 3D hand model used by the authors consists of 5 *open kinematic chains*. Each chain is rooted in the palm of the hand, which has 6 degrees of freedom. From the root, each chain is built from three segments for each of the five digits. The thumb has five rotational degrees of freedom, whereas the other fingers have only four. This means that the model has in total 16 segments and 27 degrees of freedom.

The 3D hand model was morphed into different poses, and 15 model points were selected for each hand part. Inliers were generated by altering the model points with Gaussian noise (variance: 10 percent of the size of the bounding box of the data set). A number of outliers equal to 30% of the total number of model points were drawn from a uniform distribution and added to the data set.

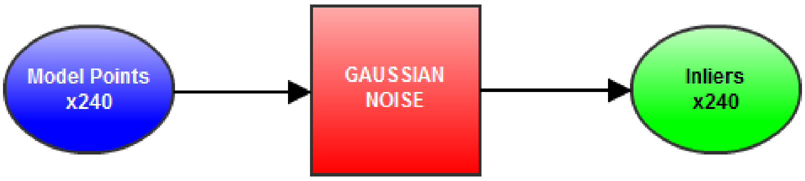

Figure 10 – ECMPR-Articulated Experiment 1 inliers.

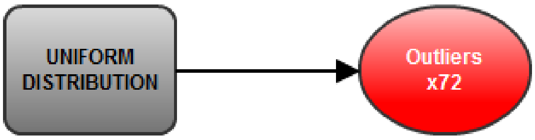

Figure 11 – ECMPR-Articulated Experiment 1 outliers.

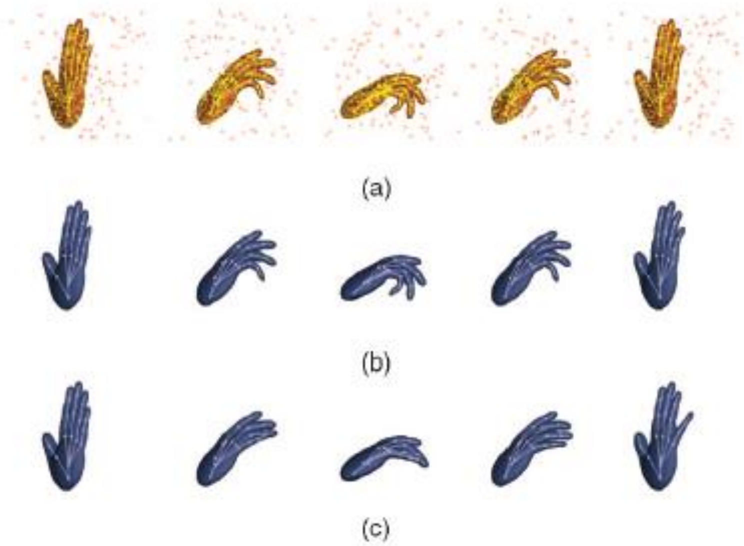

Figure 12 - (a) The ground truth of the simulated poses for ECMPR-Articulated in Experiment 1 and the simulated data (inliers and outliers). (b) A correct registration result. (c) ECMPR-Articulated failed to correctly estimate all of the kinematic parameters due to an improper initialization of the covariance matrix. This figure is a reproduction of Horaud et al., Fig. 6.

When the algorithm was initialized with large covariances, the results were excellent. Part (b) of Figure 12 illustrates this. However, when initialized with small covariances, the results were poor. The authors attribute this to the fact that a small covariance overemphasizes the importance of points quite near to the model point while essentially ignoring other points completely. This ignorance seems to reduce the quality of the results of the algorithm. Initializing with large spherical covariances increases the performance of the algorithm, but it also increases the number of iterations necessary to arrive at the optimal parameter assignment. Figure 8 in the original article (page 593) shows the performance of the algorithm by plotting digit angle vs. frame.

The authors found that the number of outliers had little impact on the performance of the algorithm. They believe that the experiment also proves the importance of using an anisotropic covariance model. One critique of the article is that the authors do not report data to show the difference between initialization with isotropic vs. anisotropic covariances, only large vs. small covariances. This makes it difficult to validate such a claim.

Experimental Setup 2

The second ECMPR-Articulated experiment was performed on real data.

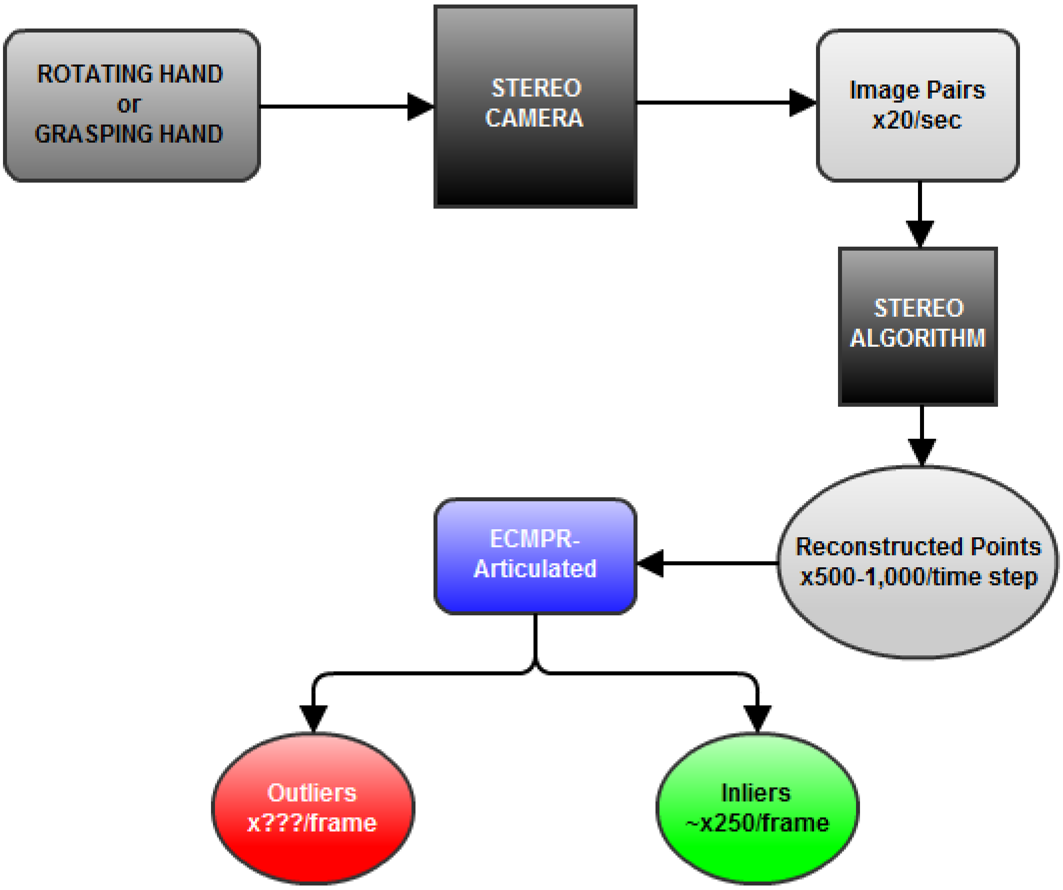

Figure 13 - ECMPR-Articulated Experiment 2 setup summary.

In this experiment, a stereo camera recorded the motion of a human hand over the course of five seconds. At 20 frames per second, this amounted to 100 image pairs for each data sequence. The data sets of 3D points are passed to ECMPR-Articulated, which recovers the transformation between the image pairs. The authors animate the 3D hand model using this transformation to visualize the efficacy of the algorithm, as seen in Figure 14. Although such visualizations are helpful, one critique of the article is the lack of a data table summarizing the quantitative performance of the algorithm. The other experiments are accompanied by tables and/or graphs, and the fact that this experiment was performed with real data makes such figures even more desirable.

Furthermore, the authors mention that between 500 and 1,000 reconstructed points are generated by the stereo algorithm per *time step*. One critique that could be made is the use of the phrase "time step." It is unclear whether the authors mean that 500-1,000 reconstructed points are generated per image pair or per second, as they give the frame rate in frames per second.

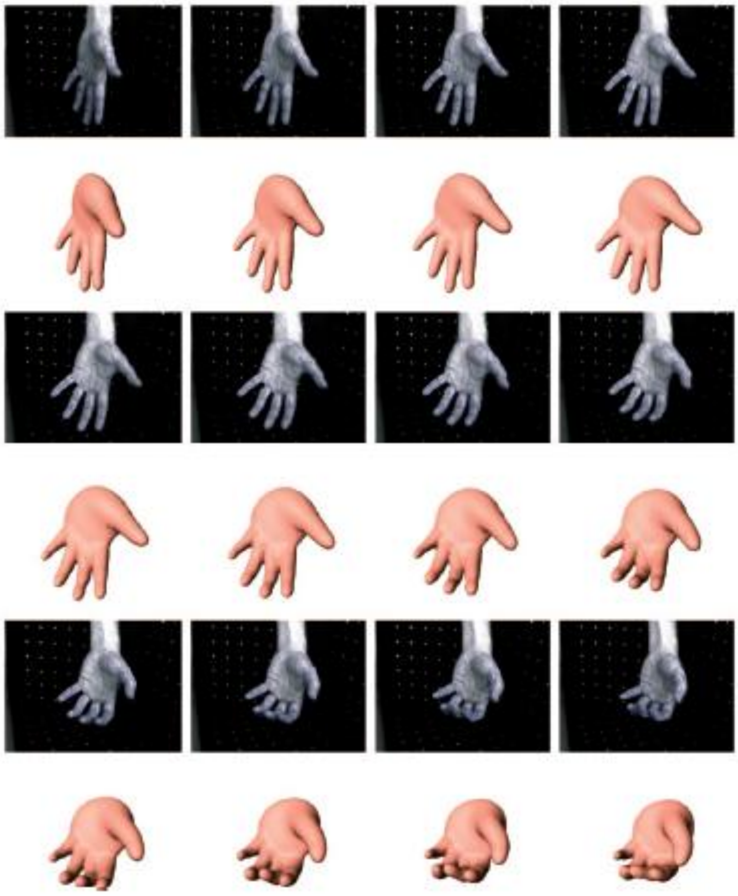

Figure 14 – The image of a hand and the result of tracking for a grasping movement. This figure is a reproduction of Horaud et al. Fig. 9.

The authors draw the paper to a close in the Conclusion section.

Conclusion

After summarizing the original contributions of their work, the authors generalize that ECMPR performs better than ICP. They support this claim by pointing out that ECMPR is less sensitive to initialization and more robust to outliers than ICP is.

The authors have a suggestion as regards the possible improvement of the efficiency of ECMPR. An implementation of the EM algorithm known as *Classification* EM (CEM) exists which forces the posterior probabilities to either 1 or 0 after each E-step. This method is suboptimal, but the authors plan to study it in the context of point registration not only to improve the speed of ECMPR, but also to develop a probabilistic interpretation of ICP. They claim that the latter is of use owing to the potential for a link between probabilistic and deterministic registration methods. They also indicate that a *k*D-tree data structure may help increase the efficiency of ECMPR at the E-steps.

As far as the potential applications of their findings, the authors have no specific comments. However, it is clear that their work has potential applications in computer vision object recognition as well as in motion tracking. Also, the 3D/3D registration of certain organs (such as those found in the respiratory tract or the vasculature) in medical images is an important part of both surgical planning and radiation therapy.

CRITIQUE

Strengths

The main conclusion of the authors is that their ECMPR algorithm is superior to the ICP algorithm. The various experiments outlined in the article have the results to support such a claim. The fact that the authors have demonstrated the efficacy of a repeatable point registration procedure which performs better than at least one leading alternative is an accomplishment.

The curves plotted for the ECMPR-Rigid experiments were effective visual conveyers of the strength of the method. The side-by-side comparison featured in Horaud et al., Fig. 8 was particularly effective.

Finally, although a reordering of the development of the equations would have made a first-pass read more informative, the article as a whole excelled at fleshing out the “big picture concepts” of both rigid point registration and articulated point registration. Even a reader unschooled in robotics would have no problem understanding the kinematic chains described in the latter case.

Weaknesses

The main experimental shortcoming of the work is the lack of numeric results reported for the ECMPR-Articulated experiment with real data. The fact that experiments with real data are inherent less repeatable than those with simulated data creates an even more urgent need to publish clear cut quantitative data. It is unlikely that this particular article would have been accepted for publication if the ECMPR-Rigid experiment with real data had also lacked numeric figures.

Another critique that could be made of these experiments is the fact that ECMPR-Rigid was compared to only one other (and non-probabilistic) algorithm. The authors mention several EM-like algorithms which incorporate features similar to the novel algorithm. It would have been interesting to see ECMPR-Rigid compared to these predecessors in addition to TriICP.

The Introduction of the article contains a large degree of background information which is not actually relevant to the contributions of the paper. For example, the third paragraph of the Introduction discusses specific details pertaining to *soft assignment* methods which do not facilitate the comprehension of any material presented thereafter. The Introduction would have been more useful had it contained more background on the EM and ECM algorithms, and possibly more details on the use of GMMs.

Although the article tends to flow nicely in terms of its manipulation of the equations, it has a habit of proceeding from section to section in a confusing way. That is, the derivations proceed from very general concepts (for example, the observed-data likelihood function $\mathcal{L}(\cdot)$) to the specific contributions of the work (the ECMPR algorithms), but it would have benefited from either cutting out a certain amount of the general concepts or introducing the more specific concepts sooner. On a first-pass reading, it is difficult to understand the motivations behind the equation manipulations unless one is extremely familiar with the ECM algorithm and the application of GGMs to missing data problems. A worthy suggestion might be to place the ECMPR-Rigid algorithm in a gray box near the beginning of the problem formulation.

Another shortcoming of the article is the way in which it cites some of its sources for equation manipulations. The most prominent offender is Equation (9) on page 591. The source cited is *Pattern and Machine Learning* by Bishop. This book contains 703 pages, and many of its sections treat the topic of the article. However, the authors of the article offer neither an explanation of the transformation yielding Equation (9) nor a specific page number in Bishop which presents it (if interested, the transformation currently under discussion appears on page 20 of Bishop).

REFERENCES

Bishop C. *Pattern Recognition and Machine Learning.* Springer, 2006.

Chen Y and Gupta M. *EM Demystified: An Expectation-Maximization Tutorial.* Department of Electrical Engineering, University of Washington. February 2010. UWEETR-2010-0002.

Horaud R, Forbes F, Yguel M, Dewaele G, Zhang, JA. Rigid and Articulated Point Registration with Expectation Conditional Maximization. IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE. March 2011. Volume 33, Issue 3. 10.1109/TPAMI.2010.94.

Kang X, Taylor RH, Armand M, Otake Y, Yau WP, Cheung PYS, Hu Y. Correspondenceless 3D-2D registration based on expectation conditional maximization. Proc SPIE 2011 March 3, 2011;7964(1):79642Z.

ECMPR-Rigid Algorithm

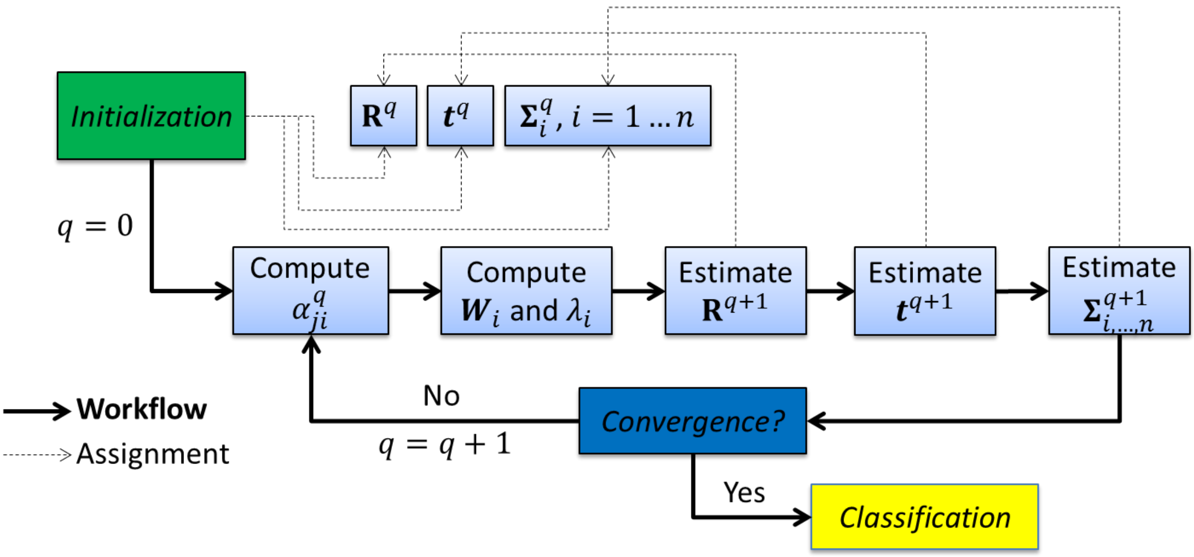

Figure 15

ECMPR-Articulated Algorithm

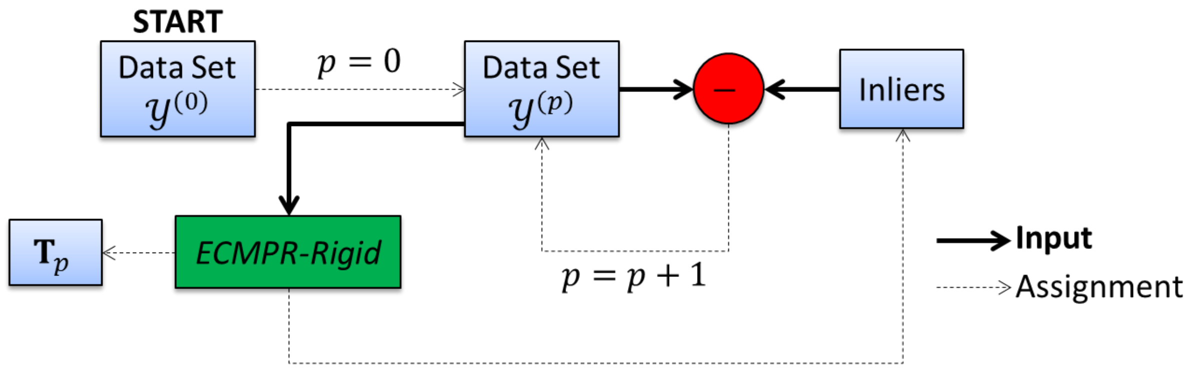

Figure 16

Expressions for Quantities Appearing in the ECMPR-Rigid Algorithm		
Quantity	Description	Formula
α_{ji}^{q}	posterior (inlier)	$\dfrac{\lvert\boldsymbol{\Sigma}_i\rvert^{-1/2}\exp\left(-\frac{1}{2}\left\Vert\boldsymbol{Y}_j-\boldsymbol{\mu}(\boldsymbol{X}_i;\boldsymbol{\Theta})\right\Vert_{\boldsymbol{\Sigma}_i}^{2}\right)}{\sum_{k=1}^{n}\lvert\boldsymbol{\Sigma}_i\rvert^{-1/2}\exp\left(-\frac{1}{2}\left\Vert\boldsymbol{Y}_j-\boldsymbol{\mu}(\boldsymbol{X}_i;\boldsymbol{\Theta})\right\Vert_{\boldsymbol{\Sigma}_i}^{2}\right)+\emptyset_{3D}}$
$\alpha_{j\,n+1}^{q}$	posterior (outlier)	$1-\sum_{i=1}^{n}a_{ji}$
$\boldsymbol{W}_i^q$	virtual observation	$\frac{1}{\lambda_i}\sum_{j=1}^{m}\alpha_{ji}\boldsymbol{Y}_j$
λ_i^q	weight of virtual observation i	$\sum_{j=1}^{m}\alpha_{ji}$
$\boldsymbol{t}^*$	$\boldsymbol{t}^{q+1}$, the new estimate of $\boldsymbol{t}^q$	$\left(\sum_{i=1}^{n}\lambda_i\boldsymbol{\Sigma}_i^{-1}\right)^{-1}\sum_{i=1}^{n}\lambda_i\boldsymbol{\Sigma}_i^{-1}(\boldsymbol{W}_i-\mathbf{R}\boldsymbol{X}_i)$
$\boldsymbol{R}^*$	$\boldsymbol{R}^{q+1}$, the new estimate of $\boldsymbol{R}^q$	$\arg\min_{\mathbf{R}}\frac{1}{2}\sum_{i=1}^{n}\lambda_i(\boldsymbol{X}_i^{\mathrm{T}}\mathbf{R}^{\mathrm{T}}\boldsymbol{\Sigma}_i^{-1}\mathbf{R}\boldsymbol{X}_i+2\boldsymbol{X}_i^{\mathrm{T}}\mathbf{R}^{\mathrm{T}}\boldsymbol{\Sigma}_i^{-1}\boldsymbol{t}^*-2\boldsymbol{X}_i^{\mathrm{T}}\mathbf{R}^{\mathrm{T}}\boldsymbol{\Sigma}_i^{-1}\boldsymbol{W}_i-2\boldsymbol{t}^{*\mathrm{T}}\boldsymbol{\Sigma}_i^{-1}\boldsymbol{W}_i+\boldsymbol{t}^{*\mathrm{T}}\boldsymbol{\Sigma}_i^{-1}\boldsymbol{t}^*)$
$\boldsymbol{\Sigma}_i^{q+1}$	the new estimate of $\boldsymbol{\Sigma}_i^q$	$\dfrac{\sum_{j=1}^{m}\alpha_{ji}^{q}\left(\boldsymbol{Y}_j-\boldsymbol{\mu}(\boldsymbol{X}_i;\boldsymbol{\Theta}^{q+1})\right)\left(\boldsymbol{Y}_j-\boldsymbol{\mu}(\boldsymbol{X}_i;\boldsymbol{\Theta}^{q+1})\right)^{\mathrm{T}}}{\sum_{j=1}^{m}\alpha_{ji}^{q}}$

Object Formatting	
$\boldsymbol{v}$	Vector v.
$\mathbf{M}$	Matrix M.
$\mathcal{Y}=\{\boldsymbol{Y}_j\}_{1\le j\le m}$	The 3D coordinates of m "observed" data points.
$\mathcal{X}=\{\boldsymbol{X}_i\}_{1\le i\le n}$	The 3D coordinates of n "model" data points.
$\mathcal{Z}=\{\boldsymbol{Z}_j\}_{1\le j\le m}$	The "missing data" (data-to-model point correspondences). $\boldsymbol{Z}_j=i$ assigns observed point $\boldsymbol{Y}_j$ to data point $\boldsymbol{X}_i$.
$\boldsymbol{\mu}:\ \mathbb{R}^3\to\mathbb{R}^3$	Transformation (applied to a model point to map it to a data point).
$\boldsymbol{\Theta}$	The set of parameters.
$\boldsymbol{\mu}(\boldsymbol{X}_i;\boldsymbol{\Theta})$	Coordinates of transformed model point (see above).
$\boldsymbol{\mu}(\boldsymbol{X}_i;\boldsymbol{\Theta})=\underset{3\text{x}3}{\mathbf{R}}\boldsymbol{X}_i+\underset{3\text{x}1}{\boldsymbol{t}},\qquad\boldsymbol{\Theta}\coloneqq\left\{\underset{3\text{x}3}{\mathbf{R}},\underset{3\text{x}1}{\boldsymbol{t}}\right\}$	Rigid transformation and its parameters.