A Collaborative Filtering Approach to Real-Time Hand Pose Estimation: Supplementary Material

Chiho Choi, Ayan Sinha, Joon Hee Choi, Sujin Jang, Karthik Ramani
Purdue University
West Lafayette, IN 47907, USA
{chihochoi, sinha12, choi240, jang64, ramani}@purdue.edu

This document serves as supplementary material to our paper 'A Collaborative Filtering Approach to Real-Time Hand Pose Estimation'. Specifically, we detail the constraints for our synthetic hand model and provide a mathematical derivation of the JMFC algorithm.

1. Hand Model

We set limits on the configuration space of the pose parameters in order to automatically generate realistic hand poses using our 3D synthetic hand model. This ensures natural hand configurations mimicking real hand gestures. A comprehensive study on the functional ranges of joint movement is conducted in [3] and [2]. We employ the Type I and II constraints articulated in these papers on our kinematic hand model with 21 DOFs. The kinematic hand model and DOFs for each joint are shown in Figure 1. The acronyms DIP, PIP, MCP, IP and TM represent distal interphalangeal joint, proximal interphalangeal joint, metacarpophalangeal joint, interphalangeal joint and trapeziometacarpal joint type, respectively. The joints with two degrees of freedom are a consequence of flexion and abduction motion.

Type I constraints set static ranges for tangible joint angle movement guided by the physical anatomy of the human hand. The angular ranges associated with the DOFs for each of the four fingers are listed in the first three rows of Table 1. Type II constraints are dynamic constraints dependent on Type I constraints. They are further subdivided into intra- and inter-finger constraints, representing the interdependence between joint angles in each finger and adjacent fingers, respectively. The intra-finger Type II joint angle constraints for all fingers, except the thumb, are listed in the last row of Table 1. The inter-finger Type II constraints limit the flexion of MCP joints in the little, ring, middle, and index fingers. For example, MCP-Flexion of the middle finger is dependent on MCP-Flexion of the index finger. Equation

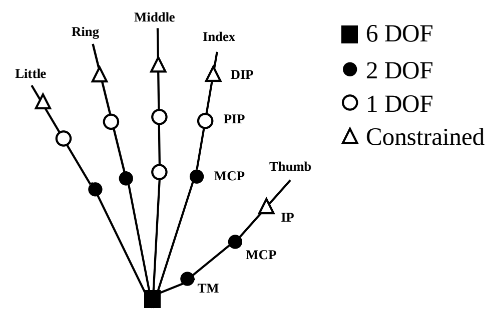

Figure 1. Our 21 DOFs hand model.

(1) iteratively governs the joint angle determination.

$$\theta^{Middle}_{MCP-F} = min(max(dmin, \theta^{Middle}_{MCP-F}), dmax), \quad (1)$$

where $dmin = max(\theta^{Index}_{MCP-F} - 25, \theta^{Ring}_{MCP-F} - 45, 0)$ and $dmax = min(\theta^{Index}_{MCP-F} + 54, \theta^{Ring}_{MCP-F} + 20, 90)$ are dynamic ranges as explained in [2]. We refer the reader to [2] for a complete list of inter-finger Type II constraints.

	Index	Middle	Ring	Little
$\theta_{MCP-Flexion}$				
	$[0°, 90°]$	$[0°, 90°]$	$[0°, 90°]$	$[0°, 90°]$
$\theta_{MCP-Abduction/Adduction}$				
	$[-15°, 15°]$	$0°$	$[-15°, 15°]$	$[-15°, 15°]$
θ_{PIP}				
	$[0°, 110°]$	$[0°, 110°]$	$[0°, 110°]$	$[0°, 110°]$
θ_{DIP}				
	$\frac{2}{3}\theta_{PIP}$	$\frac{2}{3}\theta_{PIP}$	$\frac{2}{3}\theta_{PIP}$	$\frac{2}{3}\theta_{PIP}$

Table 1. Type I and II (intra-finger) constraints for index, middle, ring, and little finger.

We now list the constraints for the thumb. The Type I ranges for θ_{MCP-F} and $\theta_{MCP-Ab/Ad}$ are $[0, 60]$ and $[-5, 5]$ respectively, whereas the ranges for θ_{TM-F} and

$\theta_{TM-Ab/Ad}$ are $[0, 60]$ and $[-15, 15]$ respectively. The intra-finger Type II constraint governing θ_{IP} in the thumb is:

$$\theta_{IP} = \frac{7}{5}\theta_{MCP-F}. \tag{2}$$

The inter-finger Type II constraints for the thumb are listed in [2].

2. The JMFC Model

In this section, we present mathematical elements of the JMFC model which factorizes the distance matrix $\mathbf{D}$ in order to complete the parameter matrix $\mathbf{P}$. We briefly review the meaning of symbols used in the main manuscript. We first retrieve k similar hand poses to the input depth map from the database using the local shape descriptor, and additionally, m hand models serve as a basis of prototype poses. Using these information, we compute the distances between the hand models in basis and the k hand postures and set these values in matrix $\mathbf{D}_1$. Also, vector $\mathbf{d_2}$ is evaluated as the distance between the models in basis and an input depth map. Next, matrix $\mathbf{P}_1$ is imputed with joint angle parameters of the k hand poses. Our goal is to estimate the unknown parameters of the input depth map, by solving the optimization equation:

$$\underset{\mathbf{A}_1,\mathbf{a}_2,\mathbf{B},\mathbf{C}}{\text{argmin}} \frac{1}{2}\left\| \begin{bmatrix} \mathbf{D}_1 \\ \mathbf{d}_2 \end{bmatrix} - \begin{bmatrix} \mathbf{A}_1 \\ \mathbf{a}_2 \end{bmatrix} \mathbf{B} \right\|_F^2 + \frac{\mu}{2}\|\mathbf{P}_1 - \mathbf{A}_1\mathbf{C}\|_F^2 + \frac{\lambda}{2}\left(\|\mathbf{A}_1\|_F^2 + \|\mathbf{a}_2\|_F^2 + \|\mathbf{B}\|_F^2 + \|\mathbf{C}\|_F^2\right). \tag{3}$$

where λ, μ are regularization terms. Figure 2 shows the matrix framework of the JMFC model.

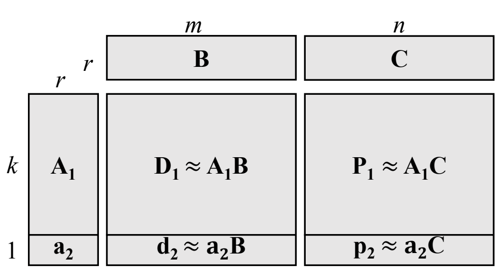

Figure 2. The matrix framework of JMFC.

We use the Alternative Least Squares (ALS) to solve the minimization problem as follows: Let the argmin of equation (3) be f. Then, the gradient of f with respect to $\mathbf{A}_1$ is:

$$\frac{\partial f}{\partial \mathbf{A}_1} = \mathbf{A}_1\left(\mathbf{B}\mathbf{B}^T + \mu\mathbf{C}\mathbf{C}^T + \lambda\mathbf{I}\right) - \left(\mathbf{D}_1\mathbf{B}^T + \mu\mathbf{P}_1\mathbf{C}^T\right) \tag{4}$$

Equating equation (4) to zero outputs the optimal solution of (3) with respect to $\mathbf{A_1}$, and is given by

$$\mathbf{A}_1 = \left(\mathbf{D}_1\mathbf{B}^T + \mu\mathbf{P}_1\mathbf{C}^T\right)\left(\mathbf{B}\mathbf{B}^T + \mu\mathbf{C}\mathbf{C}^T + \lambda\mathbf{I}\right)^{-1}. \tag{5}$$

Similarly, we can obtain the optimal solution of (3) with respect to $\mathbf{a}_2$, $\mathbf{B}$ and $\mathbf{C}$ are:

$$\mathbf{a}_2 = \left(\mathbf{d}_2\mathbf{B}^T\right)\left(\mathbf{B}\mathbf{B}^T + \lambda\mathbf{I}\right)^{-1} \tag{6}$$

$$\mathbf{B} = \left(\mathbf{A}_1^T\mathbf{A}_1 + \mathbf{a}_2^T\mathbf{a}_2 + \lambda\mathbf{I}\right)^{-1}\left(\mathbf{A}_1^T\mathbf{D}_1 + \mathbf{a}_2^T\mathbf{d}_2\right) \tag{7}$$

$$\mathbf{C} = \left(\mu\mathbf{A}_1^T\mathbf{A}_1 + \lambda\mathbf{I}\right)^{-1}\left(\mu\mathbf{A}_1^T\mathbf{P}_1\right). \tag{8}$$

We iteratively calculate (5), (6), (7) and (8) until f converges. Once f converges, we can obtain parameters of the input depth map using the equation $\mathbf{p_2} = \mathbf{a_2}\mathbf{C}$. Hence, the algorithm of our joint model is given by Algorithm 1.

Algorithm 1: The JMFC algorithm

Input: $\mathbf{D}_1, \mathbf{d}_2, \mathbf{P}_1, \mu, \lambda$
Initialize: $\mathbf{A}_1, \mathbf{a}_2, \mathbf{B}, \mathbf{C}$
while *stopping criterion not met* **do**
  $\mathbf{A}_1 \leftarrow \left(\mathbf{D}_1\mathbf{B}^T + \mu\mathbf{P}_1\mathbf{C}^T\right)\left(\mathbf{B}\mathbf{B}^T + \mu\mathbf{C}\mathbf{C}^T + \lambda\mathbf{I}\right)^{-1}$
  $\mathbf{a}_2 \leftarrow \left(\mathbf{d}_2\mathbf{B}^T\right)\left(\mathbf{B}\mathbf{B}^T + \lambda\mathbf{I}\right)^{-1}$
  $\mathbf{B} \leftarrow \left(\mathbf{A}_1^T\mathbf{A}_1 + \mathbf{a}_2^T\mathbf{a}_2 + \lambda\mathbf{I}\right)^{-1}\left(\mathbf{A}_1^T\mathbf{D}_1 + \mathbf{a}_2^T\mathbf{d}_2\right)$
  $\mathbf{C} \leftarrow \left(\mu\mathbf{A}_1^T\mathbf{A}_1 + \lambda\mathbf{I}\right)^{-1}\left(\mu\mathbf{A}_1^T\mathbf{P}_1\right)$
end
$\mathbf{p}_2 \leftarrow \mathbf{a}_2\mathbf{C}$

The latent representations (matrix $\mathbf{A}, \mathbf{B}, \mathbf{C}$) are randomly initialized by uniformly sampling between 0 and 1. We contend the accuracy of JMFC will improve when initialized with PCA, albeit with a computational overhead. Instead, we propose to use improved initialization methods such as Random Acol [1] to improve JMFC in future work. The difference between PCA and our method is that our method simultaneously uses $\mathbf{D1}$ and $\mathbf{d_2}$ to obtain $\mathbf{B}$ (as opposed to PCA using only $\mathbf{D_1}$). Because the effect of $\mathbf{d_2}$ on B is small as $\mathbf{d_2} << \mathbf{D_1}$, the obtained solution is comparably robust to PCA. The orthonormal constraint imposed by PCA is unnecessary because, it only leads to scaling the rows of $\mathbf{A_1}$ not affecting the final outcome. Meanwhile, our method is much faster than PCA because only a few iterations (< 100) are required for convergence.

References

[1] A. N. Langville, C. D. Meyer, R. Albright, J. Cox, and D. Duling. Initializations for the nonnegative matrix factorization. In *Proceedings of the twelfth ACM SIGKDD international conference on knowledge discovery and data mining*, pages 23–26. Citeseer, 2006. 2

[2] J. Lee and T. Kunii. Model-based analysis of hand posture. *Computer Graphics and Applications, IEEE*, 15(5):77–86, Sep 1995. 1, 2

[3] J. Lin, Y. Wu, and T. S. Huang. Modeling the constraints of human hand motion. In *Proceedings of the Workshop on Human Motion*, pages 121–, Washington, DC, USA, 2000. 1