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 (1) iteratively governs the joint angle determination.

\[ \theta_{MCP-F}^{\text{Middle}} = \min(\max(d_{\text{min}}, \theta_{MCP-F}^{\text{Middle}}), d_{\text{max}}), \] (1)

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

<table>
<thead>
<tr>
<th>Finger</th>
<th>Flexion</th>
<th>Abduction/Adduction</th>
<th>PIP</th>
<th>DIP</th>
</tr>
</thead>
<tbody>
<tr>
<td>Index</td>
<td>[0°, 90°]</td>
<td>[−15°, 15°]</td>
<td>[0°, 110°]</td>
<td>(\frac{2}{3}\theta_{\text{PIP}})</td>
</tr>
<tr>
<td>Middle</td>
<td>[0°, 90°]</td>
<td>[−15°, 15°]</td>
<td>[0°, 110°]</td>
<td>(\frac{2}{3}\theta_{\text{PIP}})</td>
</tr>
<tr>
<td>Ring</td>
<td>[0°, 90°]</td>
<td>[−15°, 15°]</td>
<td>[0°, 110°]</td>
<td>(\frac{2}{3}\theta_{\text{PIP}})</td>
</tr>
<tr>
<td>Little</td>
<td>[0°, 90°]</td>
<td>[−15°, 15°]</td>
<td>[0°, 110°]</td>
<td>(\frac{2}{3}\theta_{\text{PIP}})</td>
</tr>
</tbody>
</table>

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 \(\theta_{MCP-F}\) and \(\theta_{MCP-Ab/Ad}\) are [0, 60] and [−5, 5] respectively, whereas the ranges for \(\theta_{TM-F}\) and

Figure 1. Our 21 DOFs hand model.
The latent representations (matrix A, B, 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 D1 and d2 to obtain B (as opposed to PCA using only D1). Because the effect of d2 on B is small as d2 << D1, 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 A1 not affecting the final outcome. Meanwhile, our method is much faster than PCA because only a few iterations (<< 100) are required for convergence.

References
