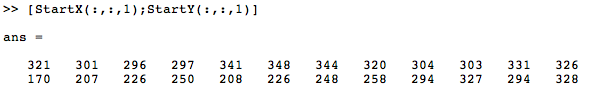
**MATLAB Codes for Producing Pairwise Shuffled Motion (PSM) sequences**

**June 14, 2015**

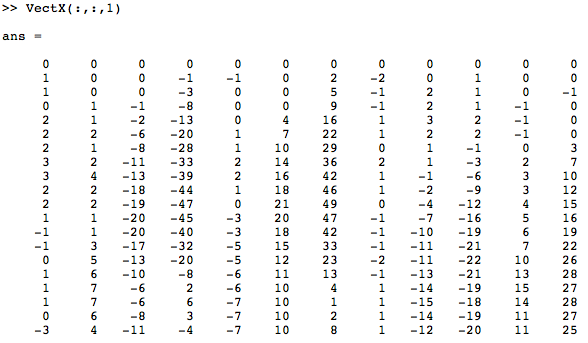
The following summarizes the MATLAB files for generating PSM animations. All m-file scripts were written in MATLAB version 8.1 (R2013a).

**1. BM\_Lib.mat**

This file is an unperturbed BM library. It contains 1) the initial positions of 12 dots of 6 exemplar actions (StartX, StartY) and 2) the values of dot position change relative to the initial X,Y coordinate, frame-by-frame, for the 6 actions. For instance,



represents the initial X,Y coordinates (for 12 dots) for the 1st action in the library.



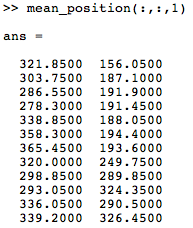
This 20 ✕ 12 array contains the values of 12 dot X-position change relative to the initial positions for the 20 frames.

**2. Get\_MeanPosition.m**

The Get\_MeanPosition.m loads BM\_Lib.mat to generate the centroid locations of each of the 12 dot trajectories for all actions in the library. Using the StartX, StartY, VectX, and VectY from the BM library, the actual X,Y coordinates for each frame are calculated first, and then the arithmatic means of the 20 (frames) X,Y coordinates are calculated. The results are saved into “MeanPosition.mat”.

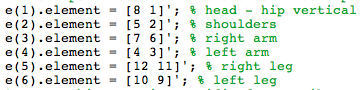
2.1. MeanPosition.mat

This .mat file has centroid (mean) coordinates of the trajectories of 12 dots for all actions. The examples below show the centroid locations of each of the dot trajectories in the 4th actions of the biological motion library. The first and the second columns represent mean X and Y positions, respectively. Each row indicates each dot.



**3. Get\_PSM\_cases.m & PSM\_cases.mat**

The file Get\_PSM\_cases.m creates new ‘orders’ of the 12 dots by following the pairwise shuffling rules described in the main text. Initially, 6 pairs of the dots are defined, each corresponding to two upper body parts (torso, shoulders) and four limbs, respectively.



In the screenshot above, the numbers within the brackets indicate the original dot ID numbers (see Figure 2 in the main text). For each pair the dot order has been reversed within the designated body part (e.g. 6 indicates right elbow and 7 is for right wrist, but they are ordered 7-6, not 6-7). And each pair is labeled 1 to 6 within a struct array. That is, e(1~6).element (*Note: the dot ID is specific to the BM\_Lib.mat. The dot IDs will be different if one uses another library of x/y/t dot positions*).

1. Next, these 6 pairs are shuffled by the pairwise shuffling rules described in the text. There are 80 different shuffling possibilities per each animation. Also note that each pair consists of two dots, therefore this shuffling rouotine works for PLs consisting of even number of dots. The code lines in Get\_PSM\_cases.m for shuffling 6 pairs with the rules are shown below. (We are currently working on a more general implementation of PSM that can manage PL animations with an odd number of dots.)

% shuffle 6 pairs with constraints

PSMcase = perms(1:6);

delidx = find(PSMcase(:,1) == 1 | PSMcase(:,1) == 2);

PSMcase(delidx,:) = [];

delidx = find(PSMcase(:,2) == 1 | PSMcase(:,2) == 2);

PSMcase(delidx,:) = [];

delidx = find(PSMcase(:,3) == 3 | PSMcase(:,3) == 4);

PSMcase(delidx,:) = [];

delidx = find(PSMcase(:,4) == 3 | PSMcase(:,4) == 4);

PSMcase(delidx,:) = [];

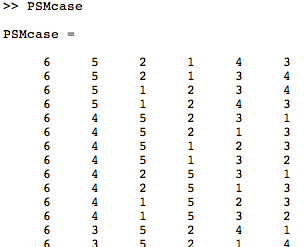
delidx = find(PSMcase(:,5) == 5 | PSMcase(:,5) == 6);

PSMcase(delidx,:) = [];

delidx = find(PSMcase(:,6) == 5 | PSMcase(:,6) == 6);

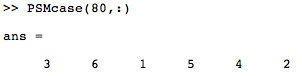
PSMcase(delidx,:) = [];

The resulting PSMcase variable has 80 different orders of the pairs.

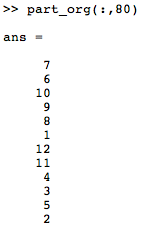


This shows part of the list within the PSMcase variable generated while running Get\_PSM\_cases.m. Each row represents reordered 6 pairs (elements). Following this, the original dot IDs are assigned to the corresponding pairs. The result from ID assignment is saved as psm\_start in “PSM\_cases.mat”.

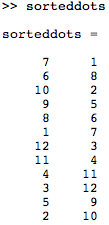
Here are some more details for creating ‘psm\_start’:



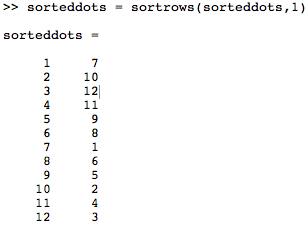
The screen above shows the last (80th) possible order of ‘pairs (element)’. According to this pair order, the dots are assigned as shown below.



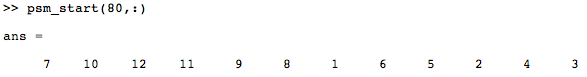
Note that the first two numbers of the array (7,6) correspond to the element ‘3’. The other dots are also listed by the pair order.



Next, the second column is added. The second column represents individual dots listed by the normal, ascending “pair” order (1,2,3,4,5,6). Keep in mind that the pair (element) “1” contains two dots (1 and 8). By comparing the two columns, one can find that the right arm (7,6) was swapped with the torso (1,8).



Next, the array is sorted based on the ascending order of the first column. The numbers in the second column are reordered accordingly.



And the second column is saved separately as the individual dot order of the 80th way of pairwise shuffling, which replaces the original dot order of the unperturbed BM (1,2,3,…12). In this way, the total 80 different versions of PSM dot order are saved as “psm\_start” into PSM\_cases.mat. That is, 80 different PSM can be created from a single BM.

Using the combination of these re-ordered set of the dots (PSM\_cases.mat), their actual centroid locations (MeanPosition.mat), and the local motion trajectories of each dot (VectX & VectY in the BM\_Lib.mat), the PSM animations are drawn on the Psychtoolbox Screen or in the built-in Plot in MATLAB.

**4. BM\_demo.m & PSM\_demo.m**

These two files are for demonstration of point-light BM and pairwise shuffled motion (PSM), respectively. Psychtoolbox 3.0 (http://psychtoolbox.org/) must be installed before running.

BM\_demo.m loads BM\_Lib.mat and draws the 5 point-light BM animations on the Psychtoolbox Screen.

PSM\_demo.m loads three .mat files, BM\_Lib.mat, MeanPosition.mat & PSM\_cases.mat. Using MeanPosition.mat and PSM\_cases.mat, the new dot positions of the first frame are created and they replace the dot positions of the first frame in unperturbed BM.

**Addendum: Pairwise Shuffled Motion for point-light (PL) animations consisting of odd number dots**

**June 24, 2015**

In studies using PL biological motion sequences, PL animations often consist of either 12 dots or of 13 dots. In our paper, we demonstrated how to generate pairwise shuffled motion (PSM) from 12-dot PL biological motion. It is natural to ask, then, how odd-numbered (e.g., 13 dots) PL biological motion can be transformed to PSM, considering that it is impossible to divide 13 dots into 6 parts each of which contains 2 dots. Here we describe one reasonable way to generate PSM from 13-dot PL animations.

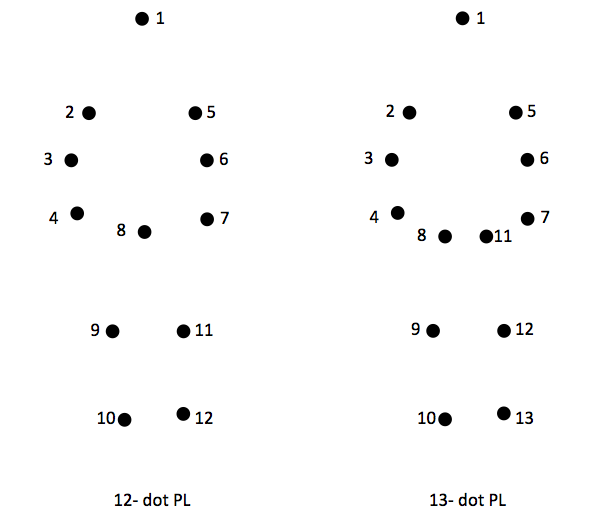


Figure 1. Exemplar frames of 12 dot (left) and 13 dot (right) PL animations

Figure 1 shows exemplar frames of 12-dot and 13-dot PL biological motion, respectively. In most cases, these two PLs are differ in the number of dots for “hip”: there are 2 dots in the 13-dot version.

To apply the PSM generation rules to a 13-dot PL sequence, we have created a virtual dot position in advance, defined the mean of the two ‘hip’ dots.

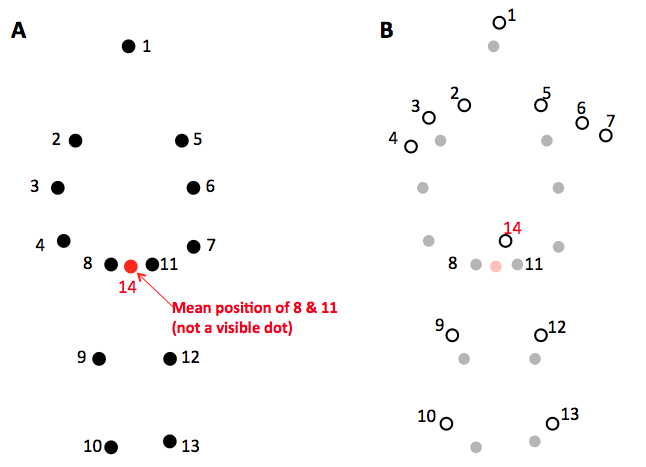


Figure 2. A: A virtual dot at the mean position of #8 and #11 (red). B: Open circles represent centroid of each dot trajectory, including the virtual dot #14.

Now, the PL (Figure 2B) is divided into 6 parts that will be swapped with each other based on the PSM generation rule (Figure 3A; also refer to the main text).

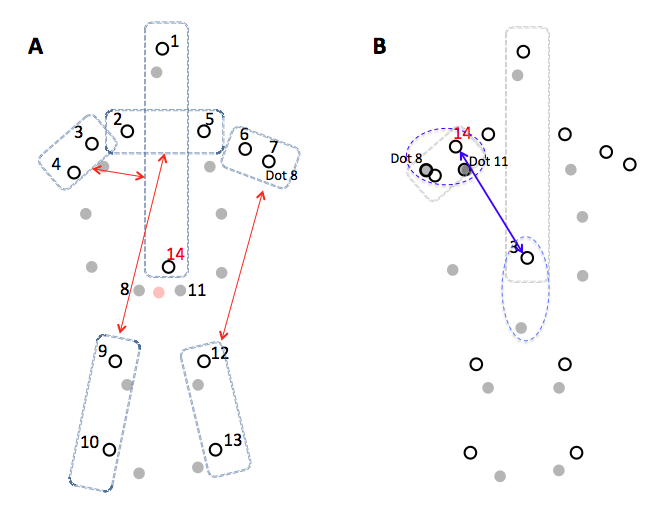


Figure 3. A. Six parts of the body for swapping with each other. Red arrows indicate one possible way of swapping out of eighty. B: Swapping #3 with the virtual position #14. When #14 moves to new location, the original dots #8 and #11 move along with #14.

Figure 3A shows 6 parts of the body and possible swapping (red arrows). Among these, Figure 3B shows swapping two dots (dot 3 and virtual dot 14) during swapping torso (dot 1 and 14) and the left arm (dot 3 and 4). Note that the original dots 8 and 11 are moved into new positions together when the virtual dot 14 moves.

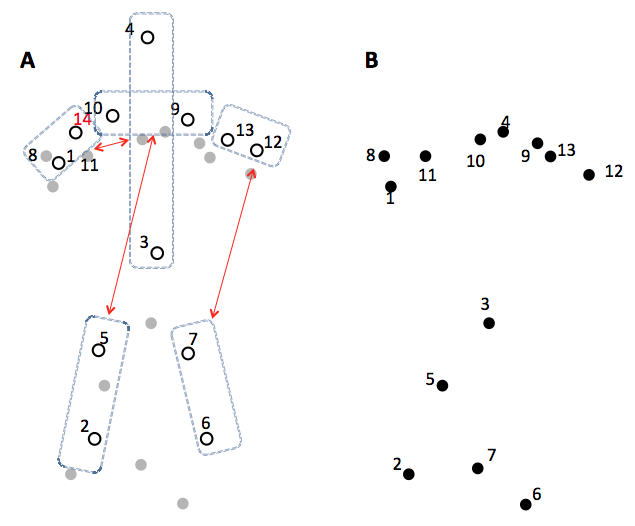


Figure 4. A: All dot swapping by the pairwise shuffling rule. B: The first frame of the PSM after shuffling.

For the other dots, each pair (2 dots) is swapped with its coupled pair (Figure 4A), which is the same as 12-dot PL shuffling. Figure 4B shows the 1st frame of the resulting PSM after shuffling.

To summarize, the rule for generating PSM for the 12-dot PL animations can be applied to the 13-dot PL animations in which two ‘hip’ dots exist. All the procedures are the same except those two ‘hip’ dots move together. Also note that the two dots are rigidly linked and not pendular/hierarchical, therefore, it is unlikely that moving the two dots together when shuffling would have an influence on the features of PSM we described in the main text.

One can see actual animations of the 13-dot walker and its 13-dot PSM counterpart by clicking [here](http://www.psy.vanderbilt.edu/faculty/blake/PSM/Movies/Demo13Dot.htm):

*Acknowledgment: We made the animation movie using the motion-capture data from CMU (mocap.sc.cmu.edu). That database was created with funding from NSF EIA-0196217. We thank Hannah Lee for her help generating the modified PSM rule described in this addendum.*