An Improved Deep Learning Architecture for Person Re-Identification: Supplementary Material

Ejaz Ahmed
University of Maryland
3364 A.V. Williams, College Park, MD 20740
ejaz@umd.edu

Michael Jones and Tim K. Marks
Mitsubishi Electric Research Labs
201 Broadway, Cambridge, MA 02139
{mjones, tmarks}@merl.com

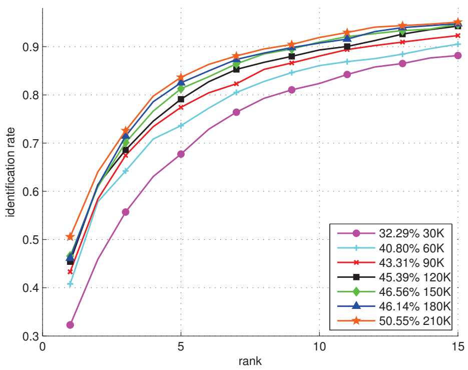

Figure 1: Performance on validation set as a function of mini-batch iterations on the CUHK03 labeled data set. In each row of the legend, the first number is the rank-1 accuracy, and the second is the number of mini-batch iterations.

1. Mini-batch Iterations and Validation Performance

Figure 1 shows the performance on the validation set as a function of mini-batch iterations on the CUHK03 labeled data set. Each mini-batch contains 100 training samples.

We also experimented with different rates of dropout after the fully connected layer. Rank-1 accuracy on the validation set for different values of dropout rate are as follows: 46.1% (no dropout), 46.9% (10% dropout), 47.1% (20% dropout), 47.6% (30% dropout), 51.3% (40% dropout) and 50.5% (50% dropout).

2. Disparity-wise Convolution

In Section 5 of the paper we discussed a few variations of our architecture. In this section we give more details about the *disparity-wise convolution* architecture. Initial layers of this architecture are the same as our proposed architecture. As in our proposed network, this architecture performs 2 tied convolutional layers each followed by max-pooling. Cross-input neighborhood differences are then computed from the features from the two views. After this step the architecture differs from the proposed architecture. Figure 2 shows the layers which differ from our proposed network. The 50 neighborhood difference maps are rearranged to give 25 groups of 50 feature maps. A convolution is then

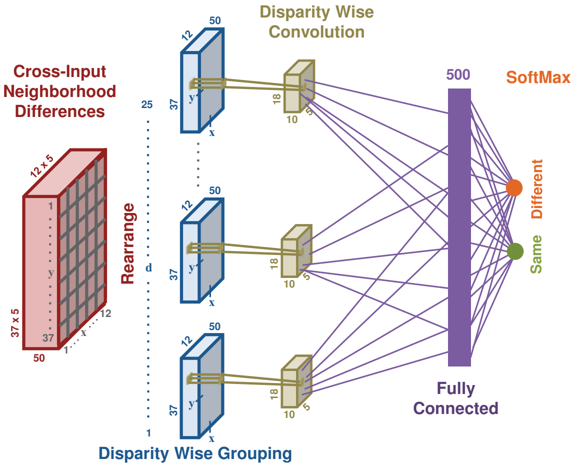

Figure 2: Disparity-wise Convolution: The initial layers are the same as our proposed architecture. Only layers that differ are shown. First, cross-input neighborhood differences are rearranged into disparity-wise groups. Each group shows feature differences at offset d. For instance, group 1 contains the values from position $(1, 1)$ of every 5×5 block in the grid of cross-input neighborhood differences, and group 25 contains the values from position $(5, 5)$ of every block in the grid. Convolution is then applied on each group separately. This is then passed through a fully connected layer and then softmax. Instead of explicitly summarizing neighborhood differences, this architecture directly learns across-patch relationships.

978-1-4673-6964-0/15/$31.00 ©2015 IEEE

applied to each of these groups followed by max-pooling. This is then passed through a fully connected layer and then softmax.

3. Qualitative Results

Figures 3, 4, and 5 show our system's ranking results on 15 randomly selected identities from the CUHK03 labeled, CUHK01 (100 identities), and VIPeR data sets, respectively. The top 25 results are sorted from left to right.

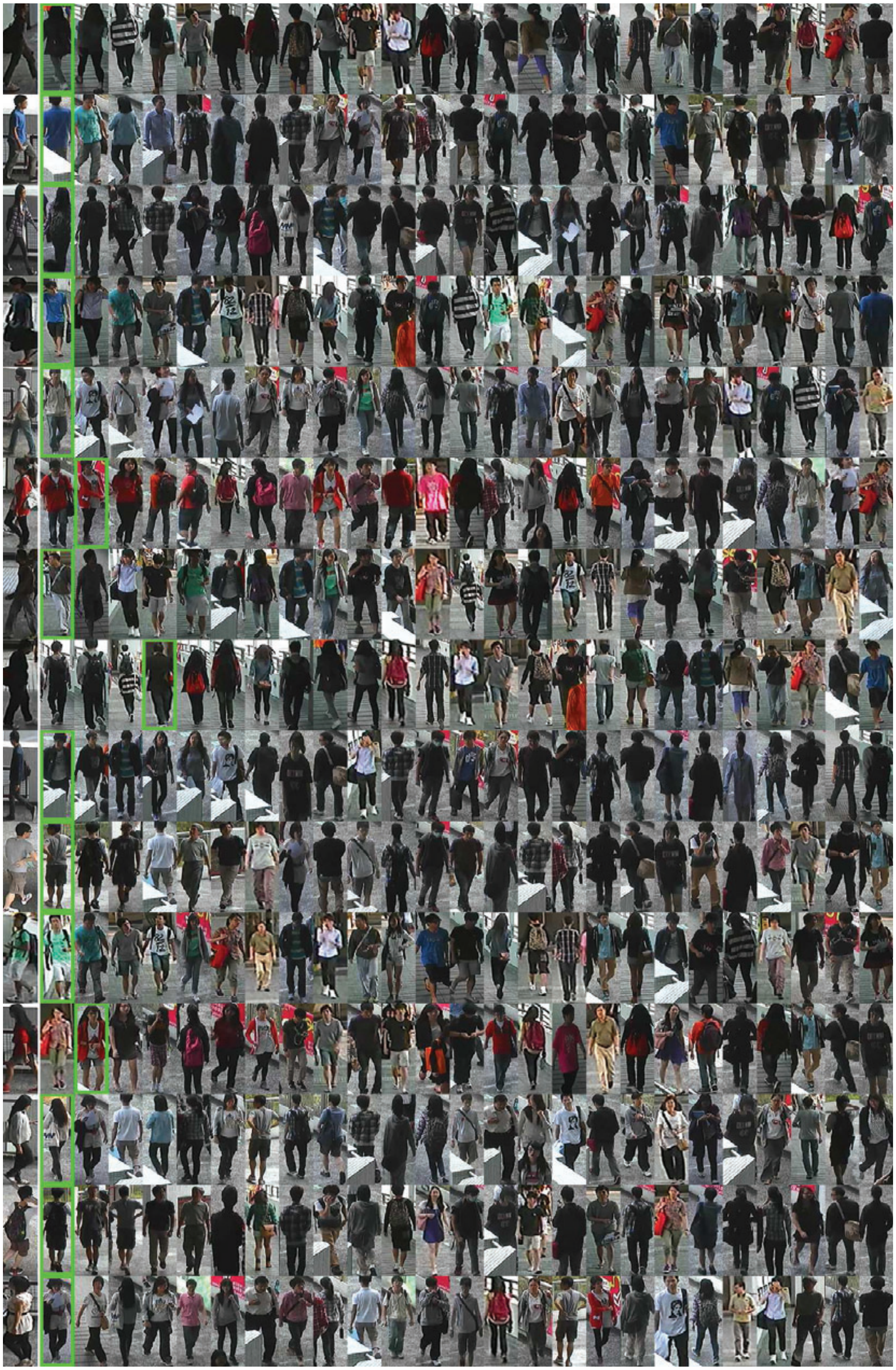

Figure 3: Example results on the CUHK03 labeled data set. In each row, the left image is the probe image, and the rest are the top 25 results sorted from left (1) to right (25). The green box indicates the correct match in each row.

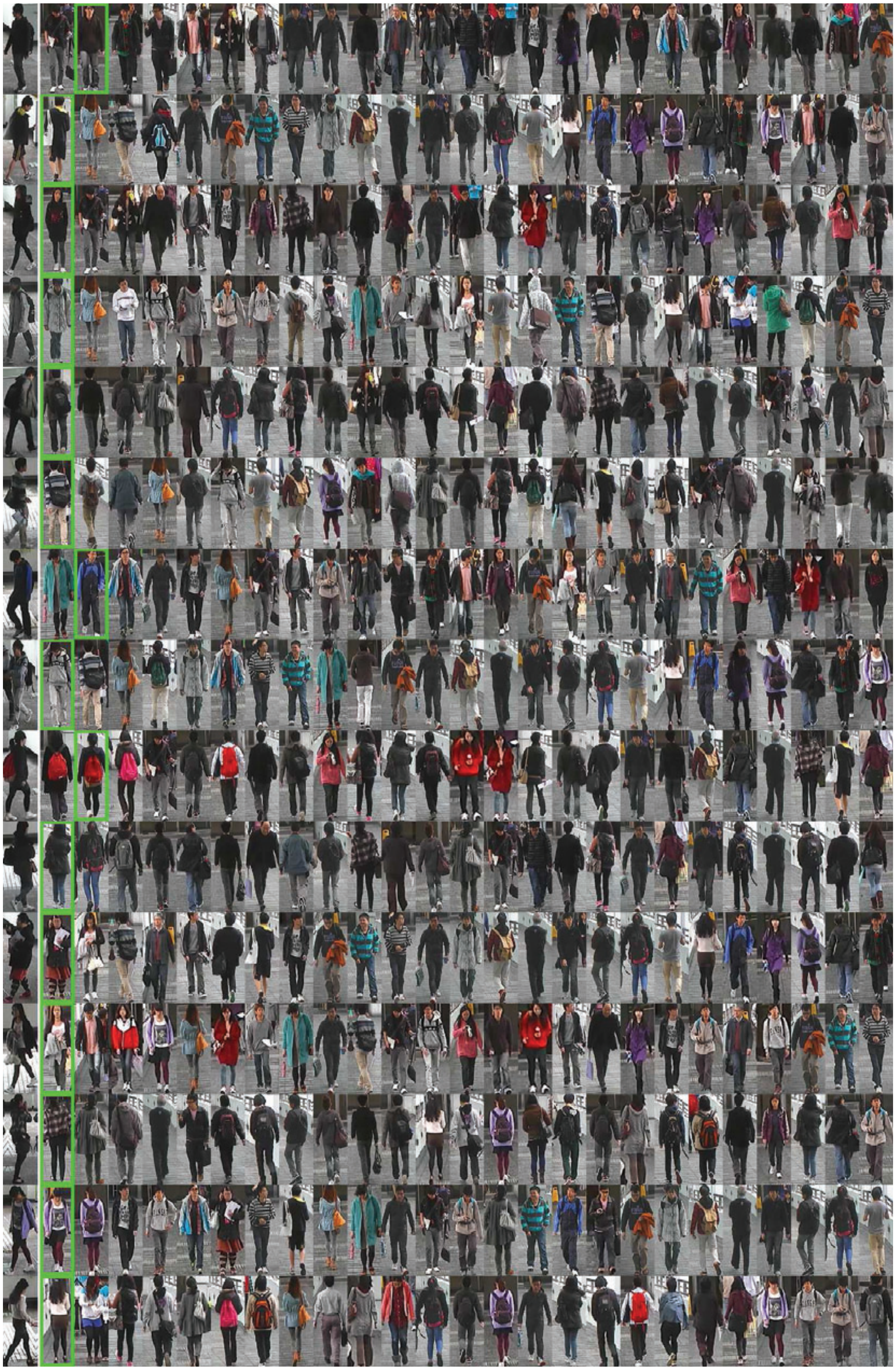

Figure 4: Example results on the CUHK01 data set (100 identities). In each row, the left image is the probe image, and the rest are the top 25 results sorted from left (1) to right (25). The green box indicates the correct match in each row.

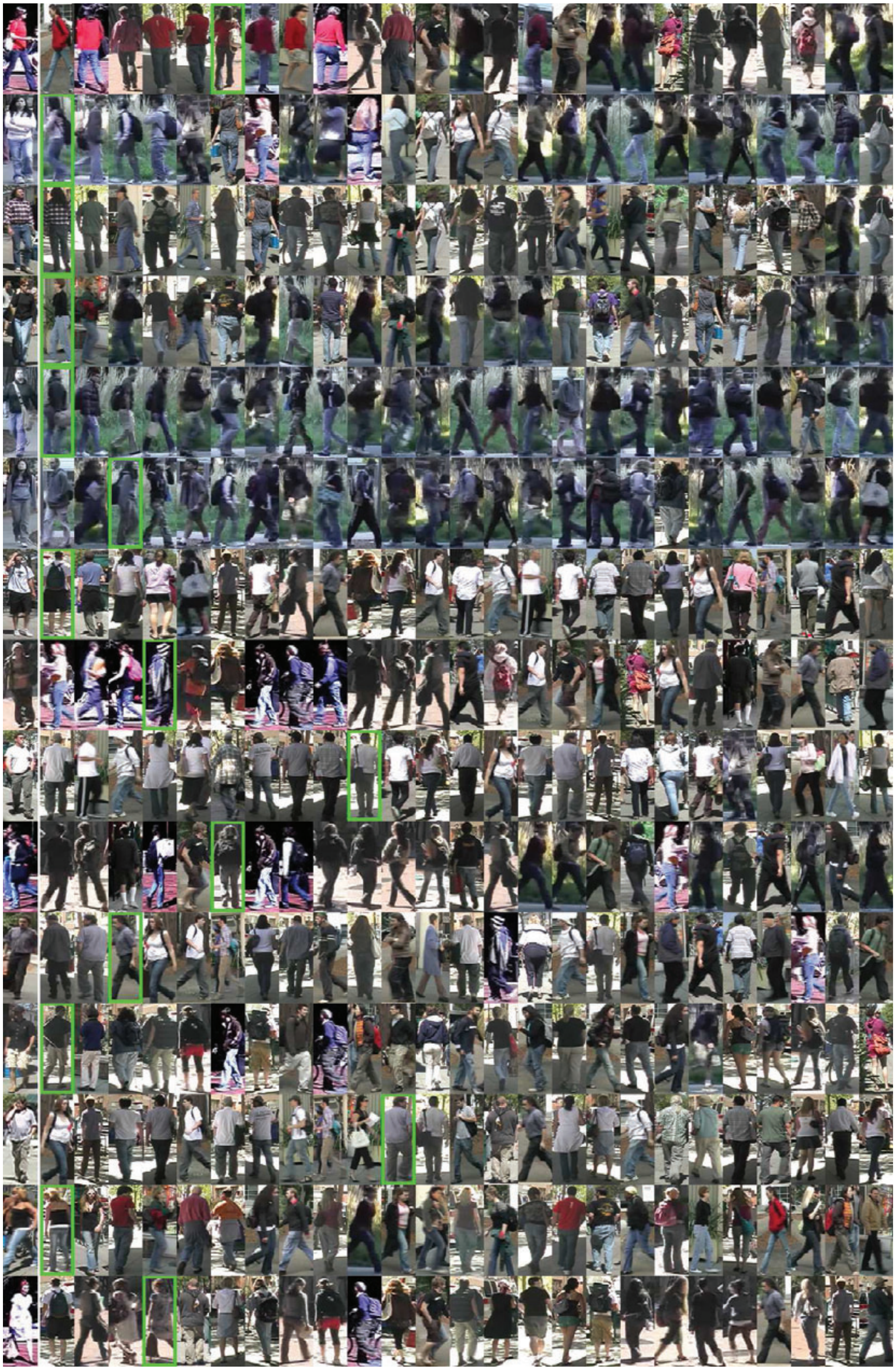

Figure 5: Example results on the VIPeR data set. In each row, the left image is the probe image, and the rest are the top 25 results sorted from left (1) to right (25). The green box indicates the correct match in each row.