# Supplementary Material
# WarpNet: Weakly Supervised Matching for Single-view Reconstruction

## 1. Computing TPS Coefficients

Given a regular grid points $\{\boldsymbol{x}_i\}$ and deformed grid points $\{\boldsymbol{x}'_i\}$, $i = 1, \dots, K^2$, the TPS transformation from the regular grid coordinate frame to the deformed grid coordinate frame for the $x$-coordinates is given by

$$T_{\theta_x}(\boldsymbol{x}) = \sum_{j=0}^{3} a^x_j \phi_j(\boldsymbol{x}) + \sum_{i=1}^{K^2} w^x_i U(||\boldsymbol{x}, \boldsymbol{x}_i||), \tag{1}$$

$$\text{s.t.} \quad \sum_{i=1}^{K^2} w^x_i = 0, \quad \sum_{j=1}^{2}\sum_{i=1}^{K^2} w^x_i x_j = 0,$$

where $\phi_0 = 1$, $\phi_j(\boldsymbol{x}_i) = x_j$, $U(r) = r^2 \log r^2$. A similar transformation may be expressed for the $y$-coordinate, denoted $T_{\theta_y}(\boldsymbol{x})$, with coefficients $\boldsymbol{w}^y$ and $\boldsymbol{a}^y$. The final transformation is $T_\theta(\boldsymbol{x}) = [T_{\theta_x}(\boldsymbol{x}), T_{\theta_y}(\boldsymbol{x})]$. With the interpolation conditions $T_\theta(\boldsymbol{x}_i) = \boldsymbol{x}'_i$, we can write the TPS coefficients $\theta = \begin{pmatrix} \boldsymbol{w}^x & \boldsymbol{w}^y \\ \boldsymbol{a}^x & \boldsymbol{a}^y \end{pmatrix}$ as the solution to a system of linear equations:

$$L\theta = \begin{pmatrix} \boldsymbol{x}' \\ 0 \end{pmatrix}, \tag{2}$$

where $L = \begin{pmatrix} K & P \\ P^T & 0 \end{pmatrix}$, $K_{ij} = U(||\boldsymbol{x}_i - \boldsymbol{x}_j||)$ and row $i$ of $P$ is $(1, x_i, y_i)$. As discussed in [2], $L$ is non-singular, invertible and only needs to be computed once, since the regular grid $\boldsymbol{x}$ is fixed for our application. Thus, computing the TPS coefficients from a deformed grid is a linear operation $\theta = L^{-1}\boldsymbol{x}'_i$ with weights, $L^{-1}$, computed once in the beginning of the training.

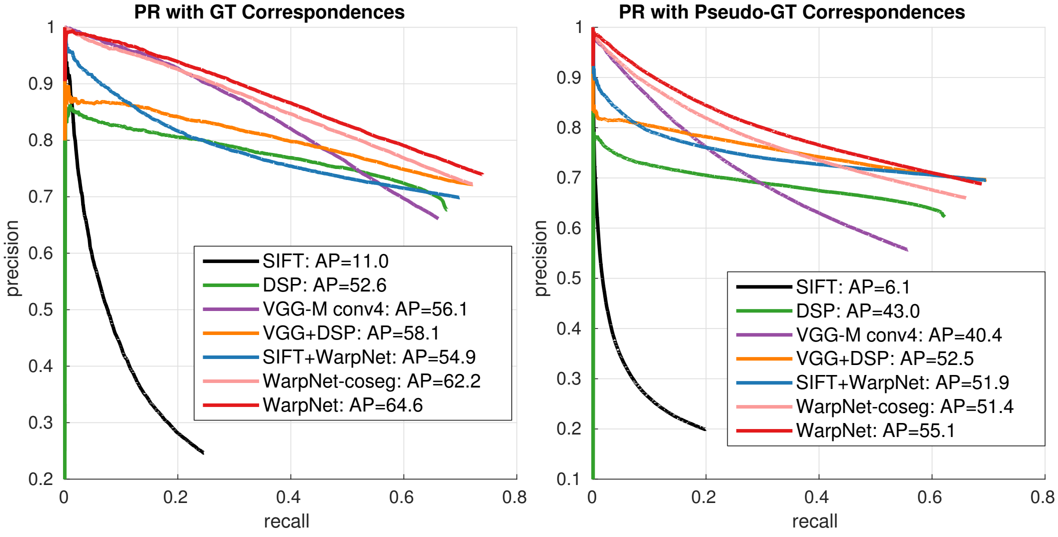
## 2. Network Training Details

**Data augmentation** Images are augmented with

- mirroring (consistent mirror for image pairs)
- scaling between $[0.8, 1.2]$
- vertical or horizontal translation by a factor within $3\%$ of image size
- rotation within $[-20, 20]$ degrees;
- contrast 1 of [1] with factors within $[0.5, 2]$
- contrast 2 of [1] with saturation multiplication factors within $[0.7, 1.4]$, saturation or hue addition within $[-0.1, 0.1]$ but with no power saturation.

All training images are cropped around the bounding box padded with 32 pixels and resized to $224 \times 224$ with pixel-wise mean subtraction computed using the CUB-200-2011 dataset.

**Training details** The feature extraction layer weights (up to `pool5`) are initialized with the VGG_M_1024 model of [3]. The learning rates on the pre-trained weights are set to one-tenth of the global learning rate. All other weights are initialized from a Gaussian distribution with a zero mean and variance equal to $0.1$.



(a) (b)

Figure 1: Precision-Recall curves for matching points between 1-nearest neighbor image pairs on the pose graph. We evaluate points with (a) human-annotated correspondences and (b) expanded pseudo-ground-truth correspondences.

We train the network with momentum $0.9$ and weight decay of $10^{-5}$. We tune the weight decay and the learning rate following the feature extraction using a held out set of artificial datasets. The learning rates of the pre-trained feature extraction layers is set to $0.01$ of the global learning rate, which is set to a constant value of $0.0001$. We train the network with mini-batch size $64$, for $45$k iterations, when the test error begins to converge.

For the point-transformer architecture (after combining `pool5` features), we experiment using several fully-connected layers instead of starting with convolution layers. However, starting with convolution layers is clearly the better choice since it yields the lowest test errors while keeping the number of parameters reasonable. We do not further fine-tune the architecture of the point-transformer such as tuning the number of feature maps, the kernel size, stride, or the number of convolution layers.

## 3. Matching Evaluation on 1-Nearest Neighbor Pose Graph

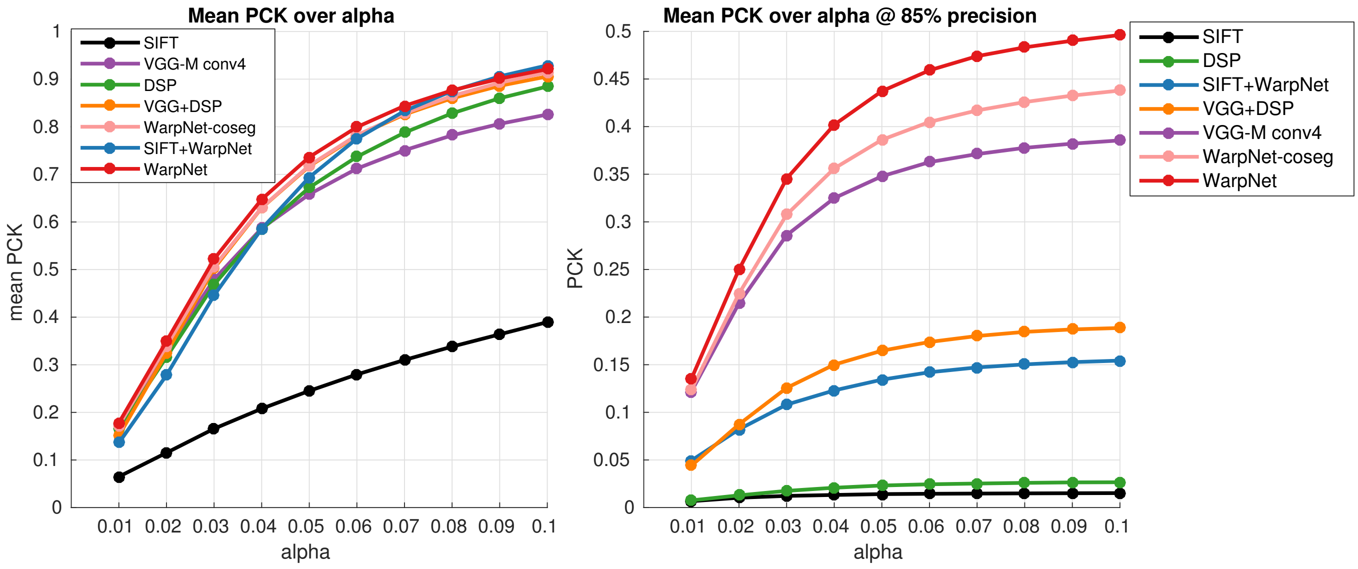
In the paper, we evaluate the matching performance on image pairs that are within 3-nearest neighbors apart on the pose graph. In this section we present evaluations on $5000$ image pairs that are 1-nearest neighbor apart on the pose graph. Figure 1 shows the precision-recall curves, analogous to Figure 8 in the paper. Figure 2 shows the PCK evaluations, analogous to Figure 9 in the paper. We observe similar trends as in the main paper, but with higher recall and PCK since 1-nearest neighbor pairs on the pose graph have less pose variations than pairs within 3-nearest neighbors.

## 4. Further Qualitative Match Comparisons

Besides the examples in the main paper, we include further qualitative match comparisons from WarpNet `conv4` and just the appearance feature of VGG-M `conv4` in Figures 3 to 10.

**Articulations** The WarpNet framework incorporates a spatial prior in the form of a thin-plate spline transformation. This lends the ability to match across articulations, typically observed in the head and tail regions of birds. Figure 3 demonstrates two such examples. Note that a CNN trained only for local appearance produces noisy matches, while WarpNet correctly accounts for the articulation.

**Viewpoint or pose variation** For reconstruction, it is crucial to be able to match across a baseline, which involves viewpoint or pose variations. The spatial prior of WarpNet affords this ability, which is difficult for the appearance-only ILSVRC CNN. In Figures 4 and 5, we show matches between instances for which both the appearance and viewpoint vary. In Figure 6, we show examples where instances have the same appearance, but different viewpoint. In each case, we observe that the WarpNet matches are robust to viewpoint variations, while the baseline NN matches are sometimes noisy. In practice, this translates into better reconstruction using the WarpNet matches.



(a)

(b)

Figure 2: PCK over varying definition of correctness $\alpha$ (higher the better). (a) Mean PCK of all retrieved matches regardless of ratio score. (b) Mean PCK with matches thresholded at $85\%$ precision, which are the matches used for reconstruction.

**Lack of texture and scale variations** Objects like birds often exhibit inter-category scale variations, or challenges such as lack of texture. By incorporating global shape information, WarpNet is able to match robustly even in the low texture regions such as the bird body, as shown in Figure 7. In Figure 8, we show examples where WarpNet can match across scale variations.

**Appearance variations** Just like the baseline CNN, WarpNet is robust to appearance variations. In Figures 9 and 10, we observe that the match quality from WarpNet is as good or better than the baseline. For non-rigid categories like birds, it is difficult to obtain instances with no viewpoint variations or articulations. So, the baseline CNN which has no spatial regularization, might produce incorrect matches that are not observed for WarpNet.

**Failure cases** While WarpNet is trained to be robust to viewpoint changes and deformations in addition to appearance variations, it may fail to produce good matches in some cases with very severe changes in viewpoint or articulation, as shown in Figure 11. Particular to objects such as birds, articulation of the head and symmetric appearance of the front and back can combine to create a situation that is difficult for matching. This is shown in Figure 12. Even though the matches seem reasonable at first glance, we observe that the back of one of the instances is matched to the belly in the other. This is somewhat mitigated for WarpNet in 12(a), but we still observe the left leg on one instance matched to the right leg on the other. We note that the appearance-only CNN also fails for these difficult cases.

## 5. Video

We include a supplementary video for better visualization of the single-view reconstruction results using four methods: `Supervised`, `WarpNet` (our method), `VGG-M` (baseline appearance-only CNN) and `DSP` (another baseline unsupervised method). We show the reconstructed shape using both depth mapping and texture mapping. The depth colors (yellow is close, blue is far) are normalized for range so that colors are comparable across all methods for each instance. In the animations, we smoothly vary the azimuth from 0° to 360°, then vary the elevation from 0° to 360°.

Similarity to the `supervised` reconstruction is best analyzed by looking at the depth coloring and observing the shape from various viewpoints. We observe that even without requiring part annotations at train or test time, `WarpNet` produces reconstructions very similar to those from the `supervised` method. In contrast, reconstructions from `VGG-M` or `DSP` exhibit noise, outliers and missing points. We note that the thin-plate spline spatial prior for `WarpNet` allows accurate reconstructions even in deformable regions such as wings, head or tail, as well as in regions of low texture such as the body.

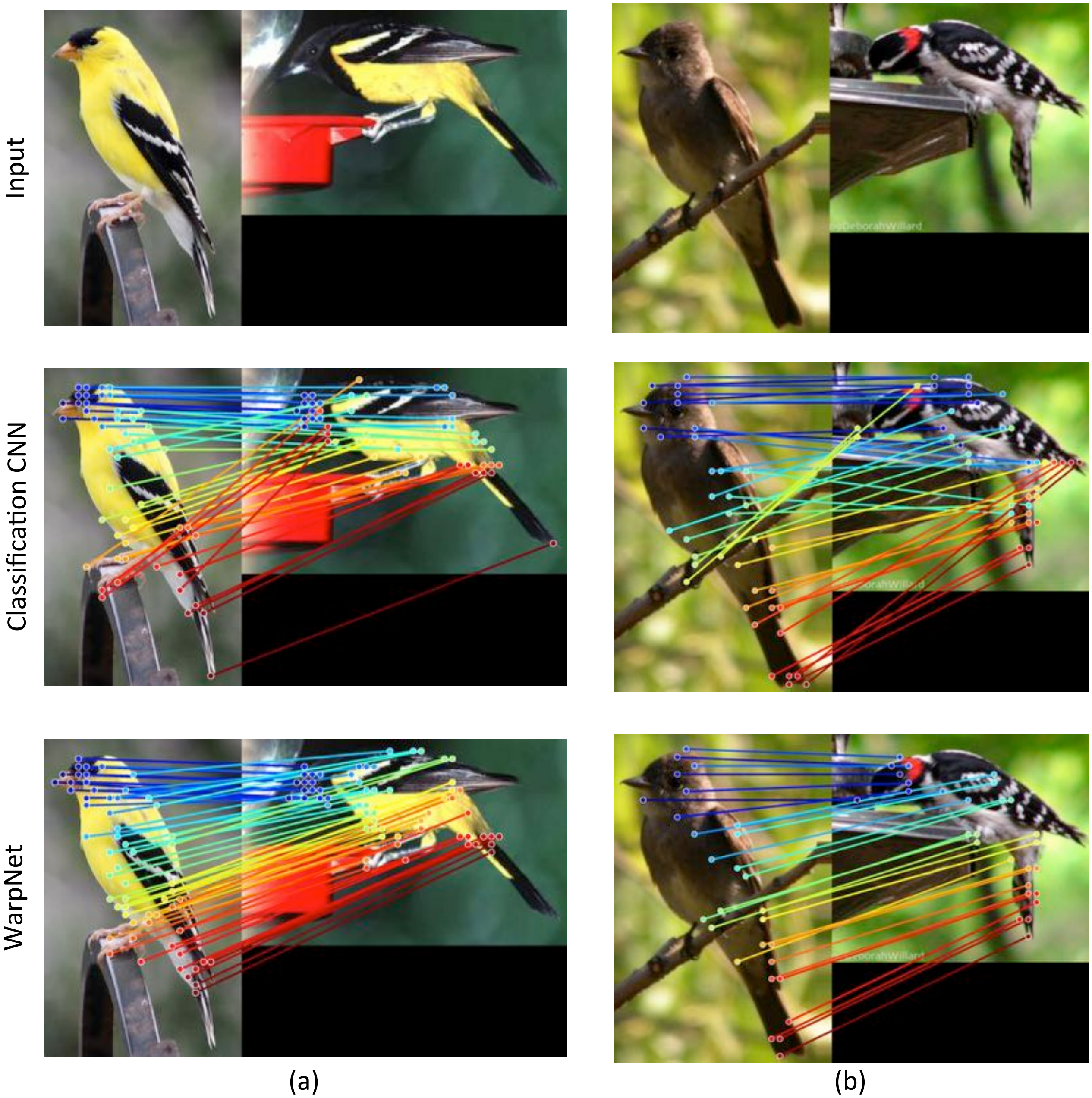


Figure 3: Qualitative match comparisons of image pairs (row 1) using the features from a CNN trained on ILSVRC (row 2) and WarpNet (row 3) for matching. Note how WarpNet can correcly match even with articulation around the tail.

## References

[1] A.Dosovitskiy, J.T.Springenberg, M.Riedmiller, and T.Brox. Discriminative unsupervised feature learning with convolutional neural networks. In *Advances in Neural Information Processing Systems 27 (NIPS)*, 2014. 1

[2] F. L. Bookstein. Principal warps: Thin-plate splines and the decomposition of deformations. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, pages 567–585, 1989. 1

[3] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman. Return of the devil in the details: Delving deep into convolutional nets. In *British Machine Vision Conference*, 2014. 1

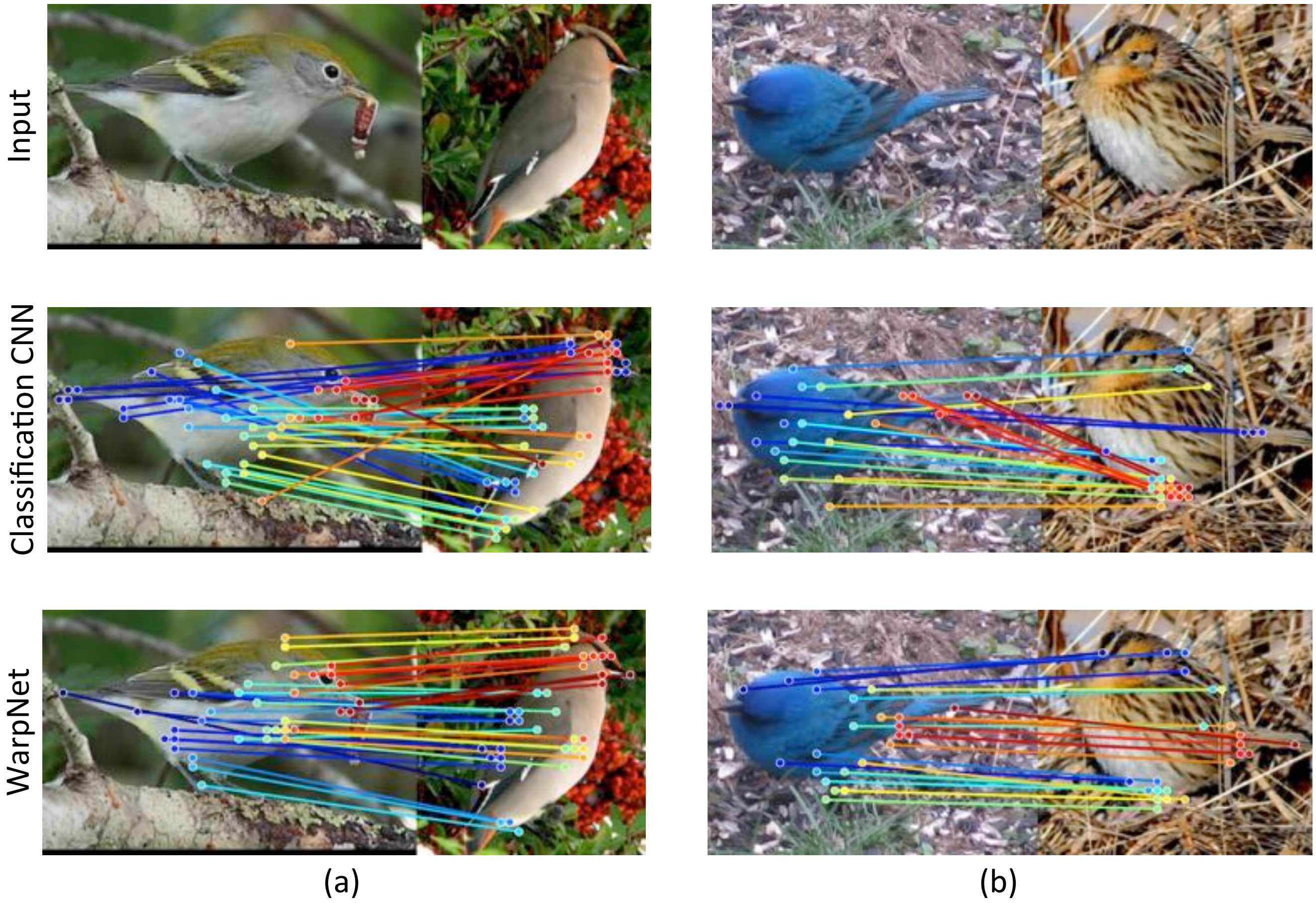


Figure 4: Qualitative match comparisons of image pairs (row 1) using the features from a CNN trained on ILSVRC (row 2) and WarpNet (row 3) for matching. Note how WarpNet obtains good matches even when both appearance and viewpoint differ.

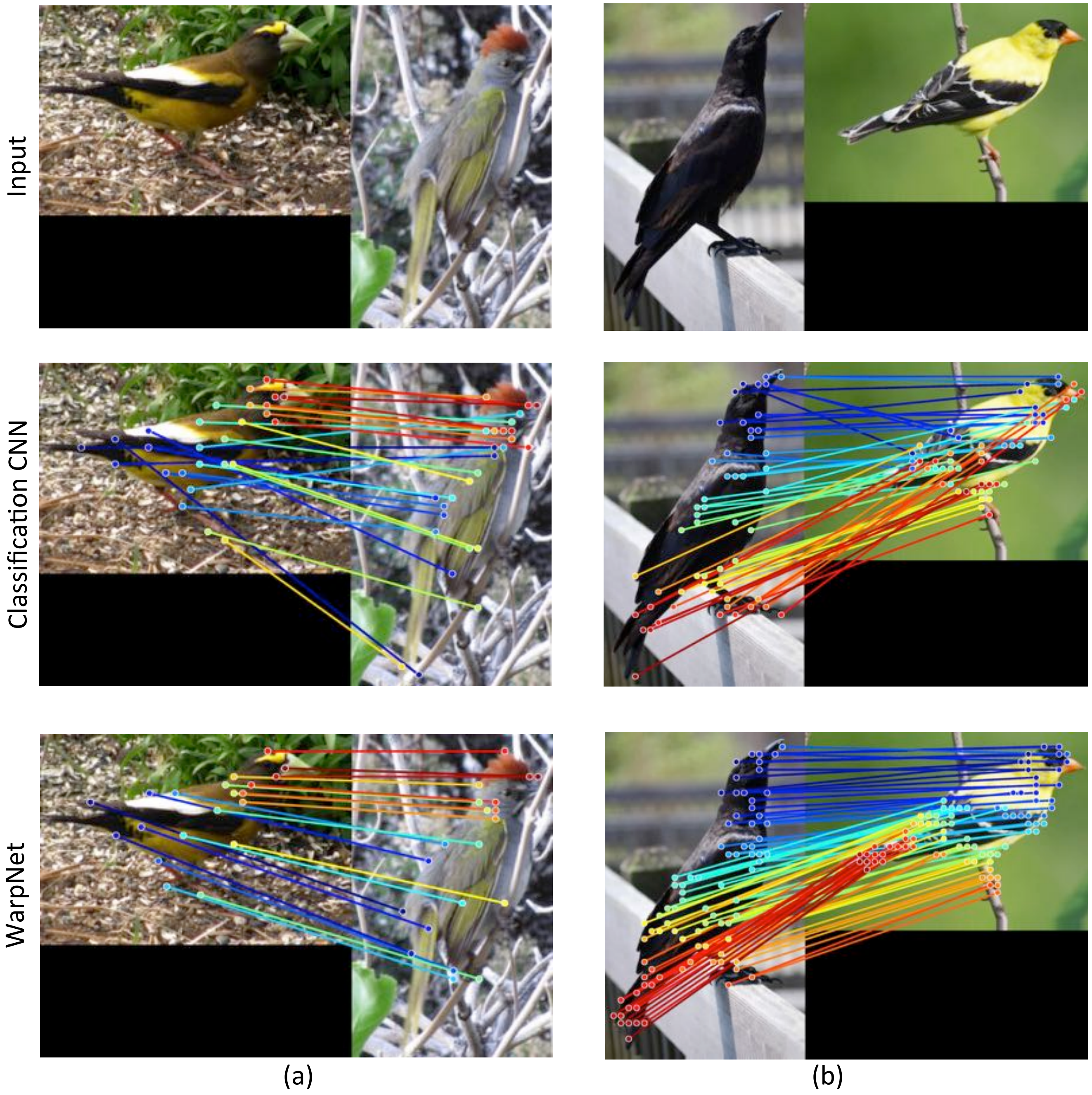


Figure 5: Qualitative match comparisons of image pairs (row 1) using the features from a CNN trained on ILSVRC (row 2) and WarpNet (row 3) for matching. WarpNet obtains good matches even when both appearance and viewpoint differ.

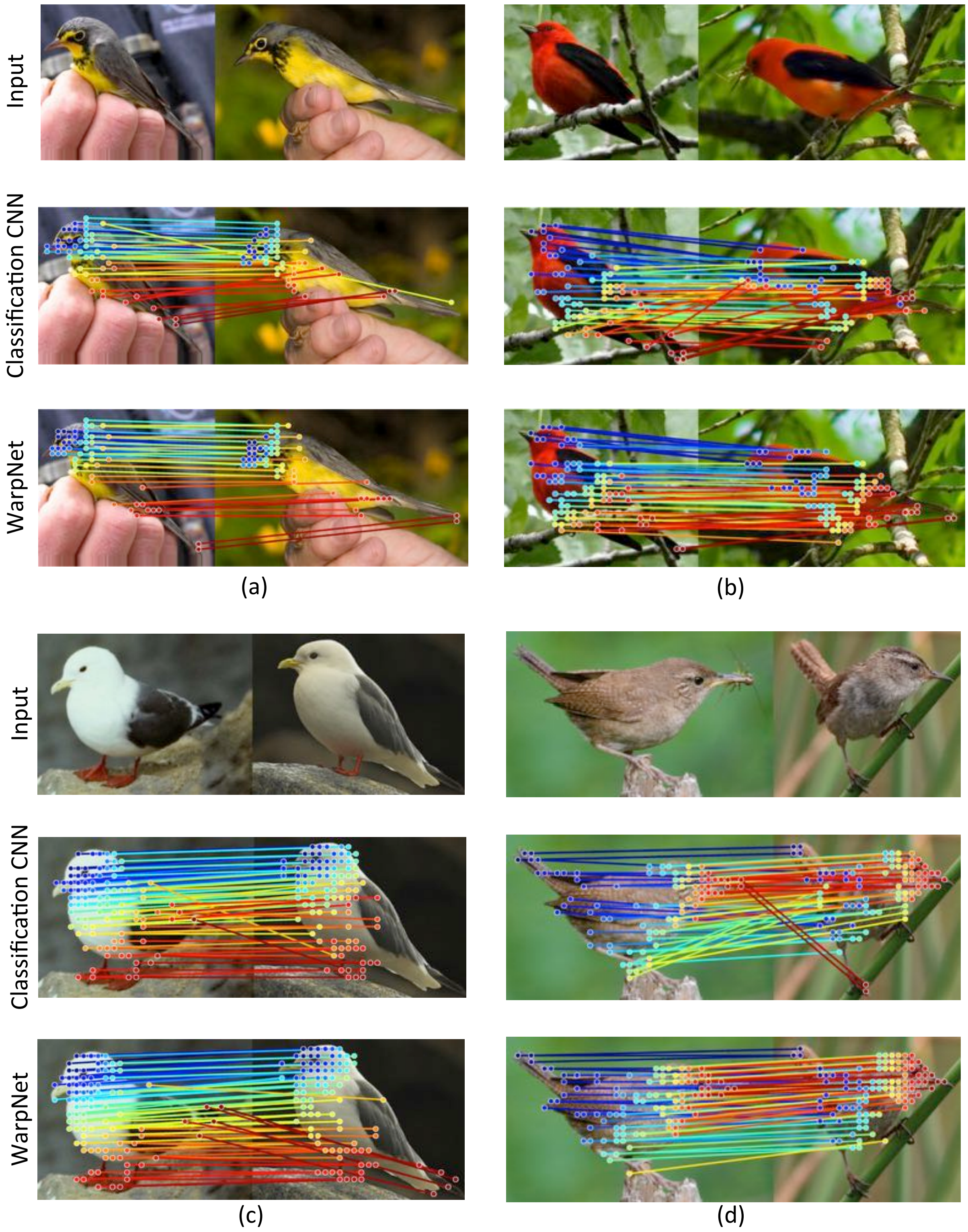


Figure 6: Qualitative match comparisons of image pairs (rows 1,4) using the features from a CNN trained on ILSVRC (rows 2,5) and WarpNet (rows 3,6) for matching. We observe that WarpNet is robust to viewpoint changes, while the baseline CNN suffers from outliers in the matching.

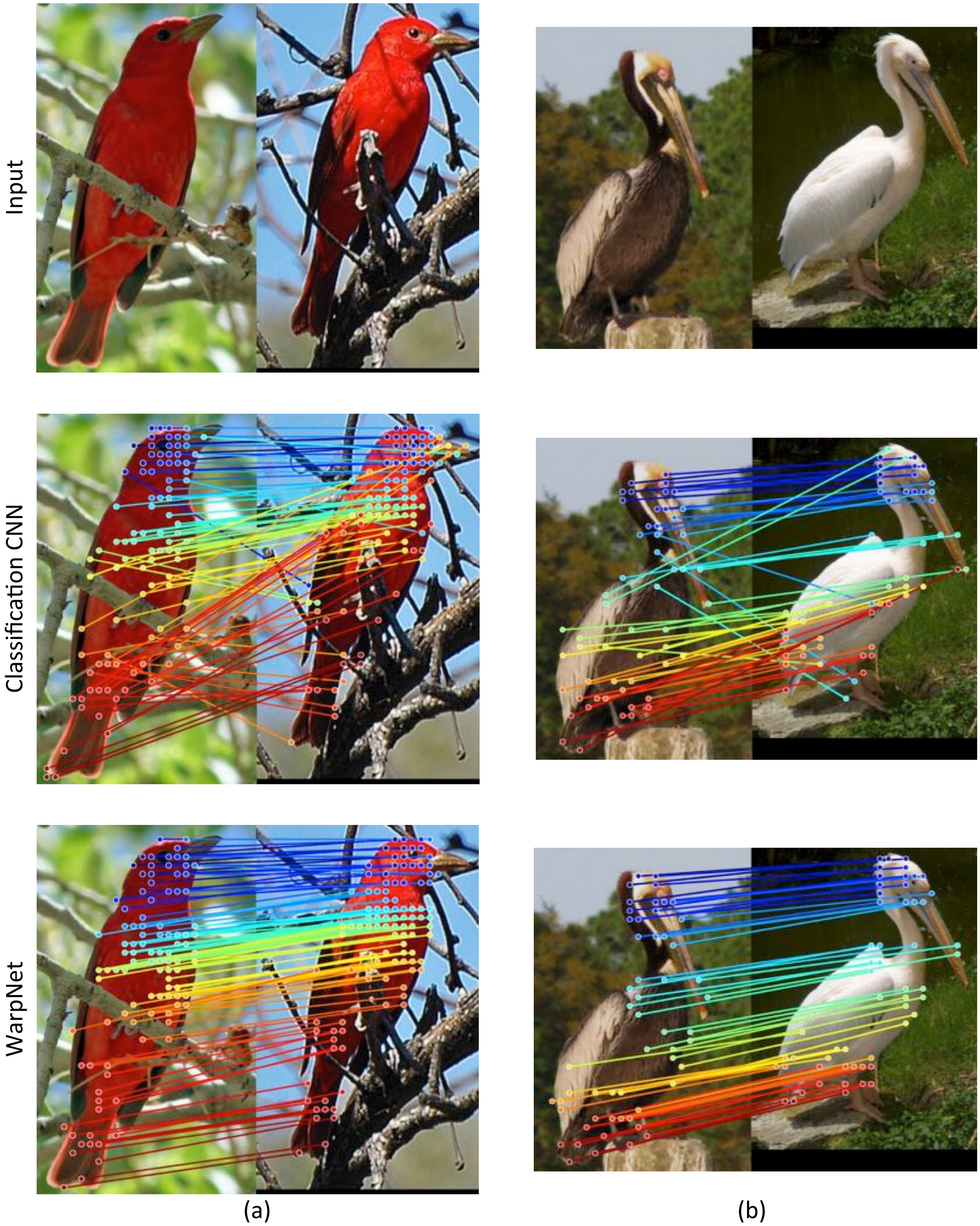


Figure 7: Qualitative match comparisons of image pairs (row 1) using the features from a CNN trained on ILSVRC (row 2) and WarpNet (row 3) for matching. Obtaining good matches using local appearance alone is challenging even when appearance and viewpoint are similar due to uniform texture of the birds.

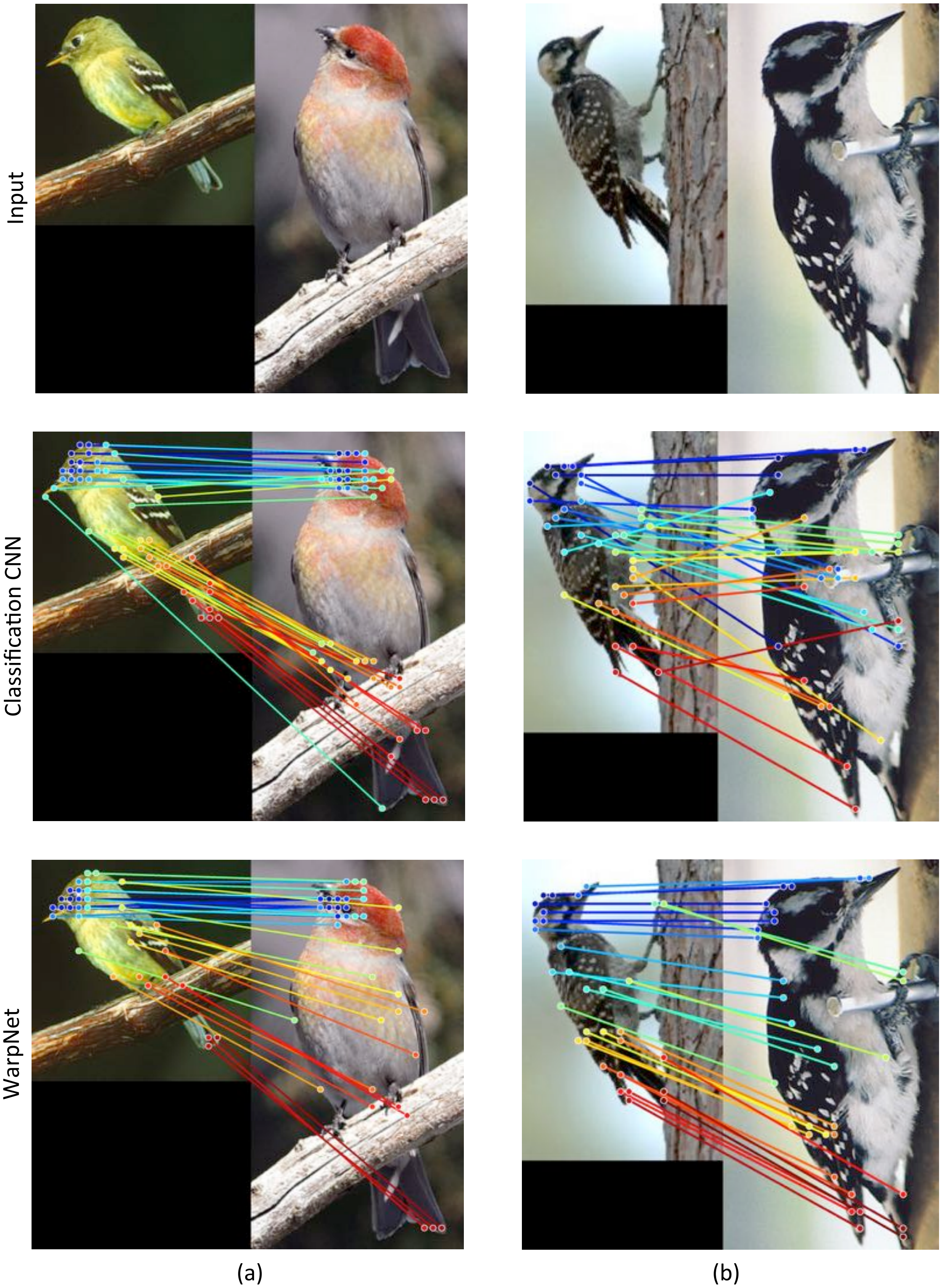


Figure 8: Qualitative match comparisons of image pairs (row 1) using the features from a CNN trained on ILSVRC (row 2) and WarpNet (row 3) for matching. We observe that WarpNet is robust to scale and appearance changes.

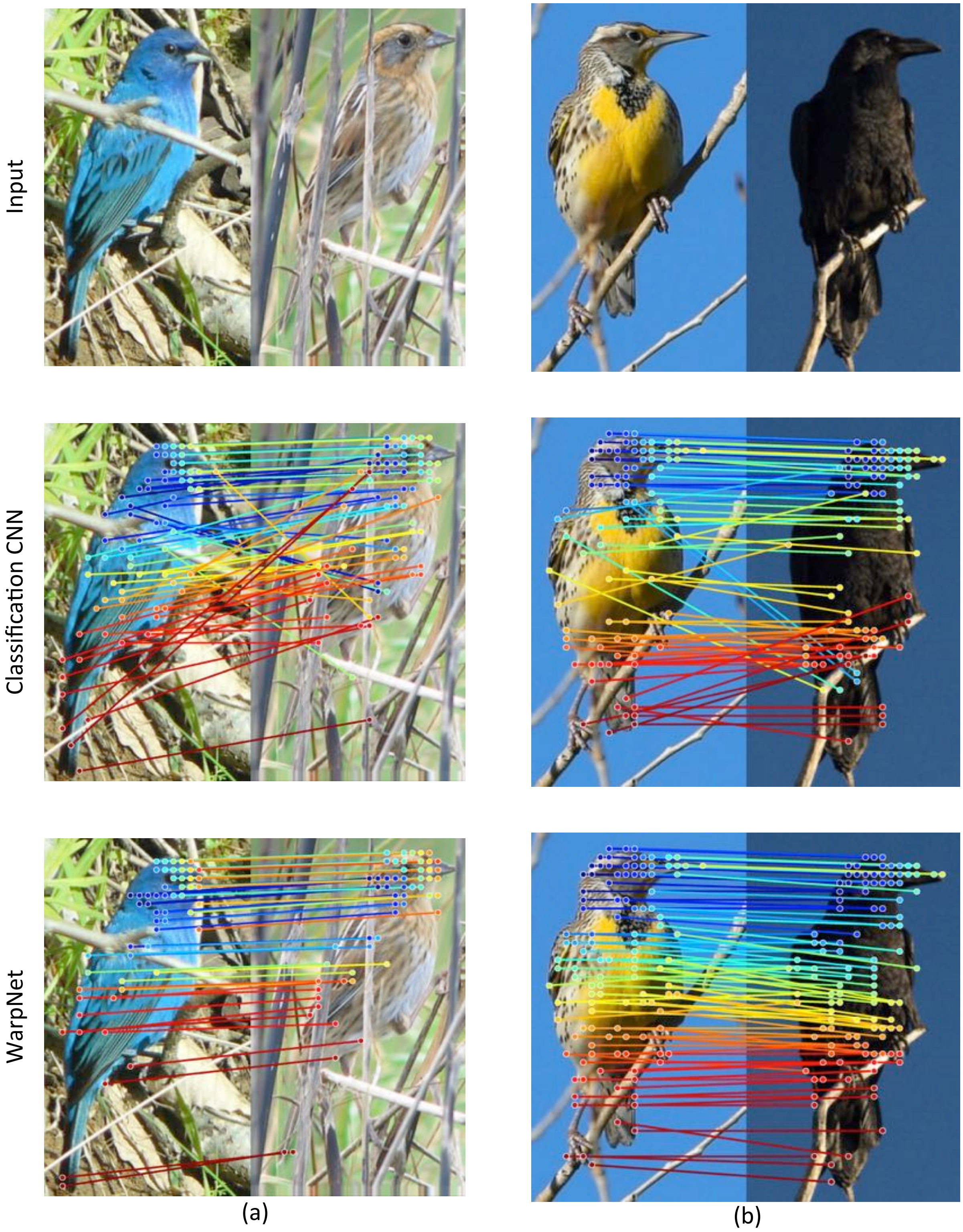


Figure 9: Qualitative match comparisons of image pairs (row 1) using the features from a CNN trained on ILSVRC (row 2) and WarpNet (row 3) for matching. WarpNet is observed to be robust to appearance variations.

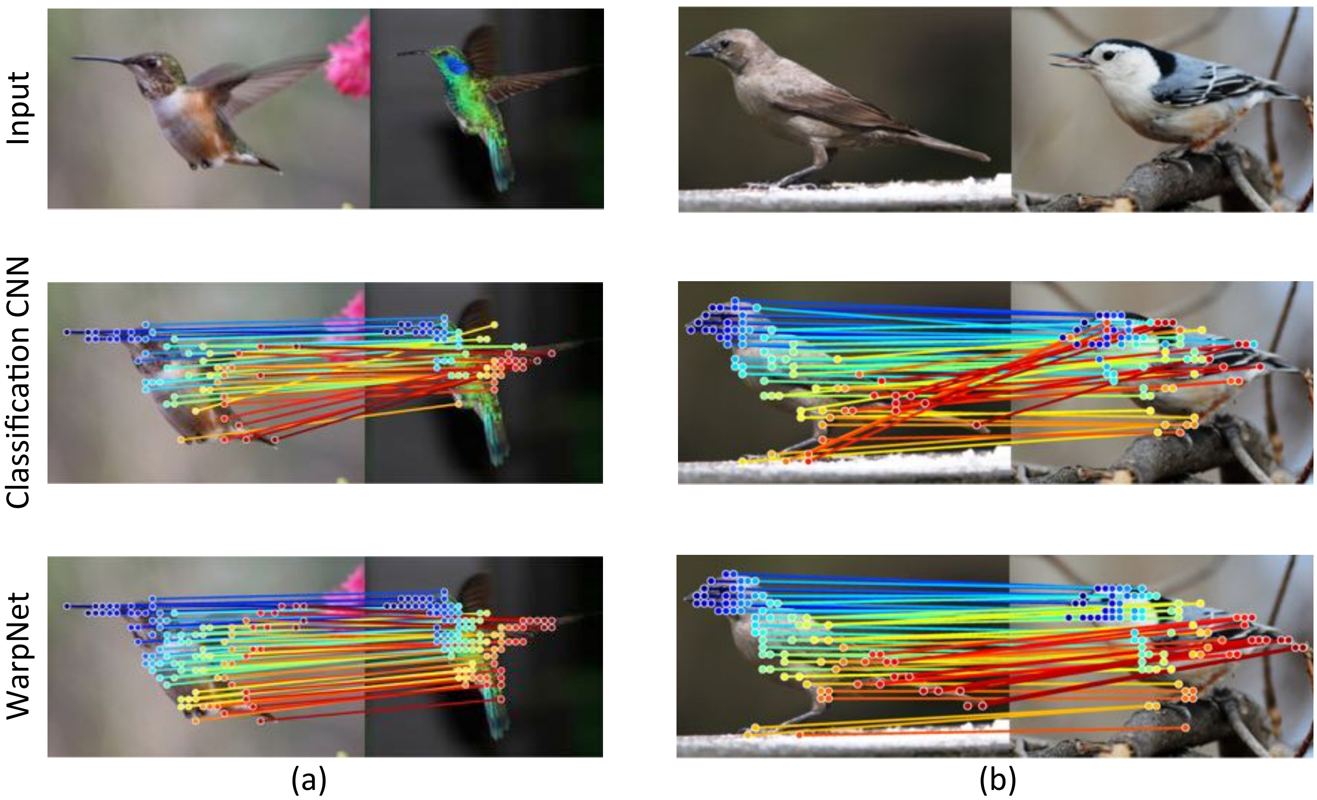


Figure 10: Qualitative match comparisons of image pairs (row 1) using the features from a CNN trained on ILSVRC (row 2) and WarpNet (row 3) for matching. Again, WarpNet demonstrates robustness to appearance variations.

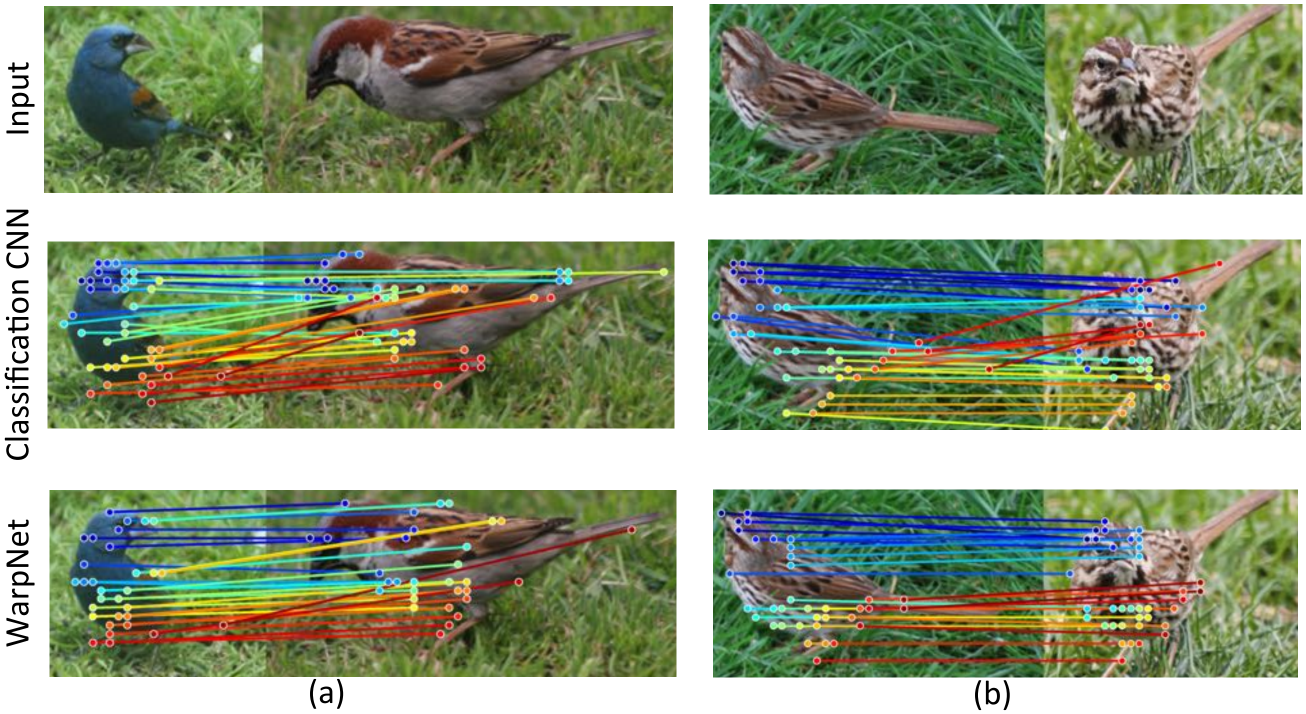


Figure 11: Failure cases for image pairs (row 1) that exhibit severe viewpoint change or articulation, using the features from a CNN trained on ILSVRC (row 2) and WarpNet (row 3) for matching.

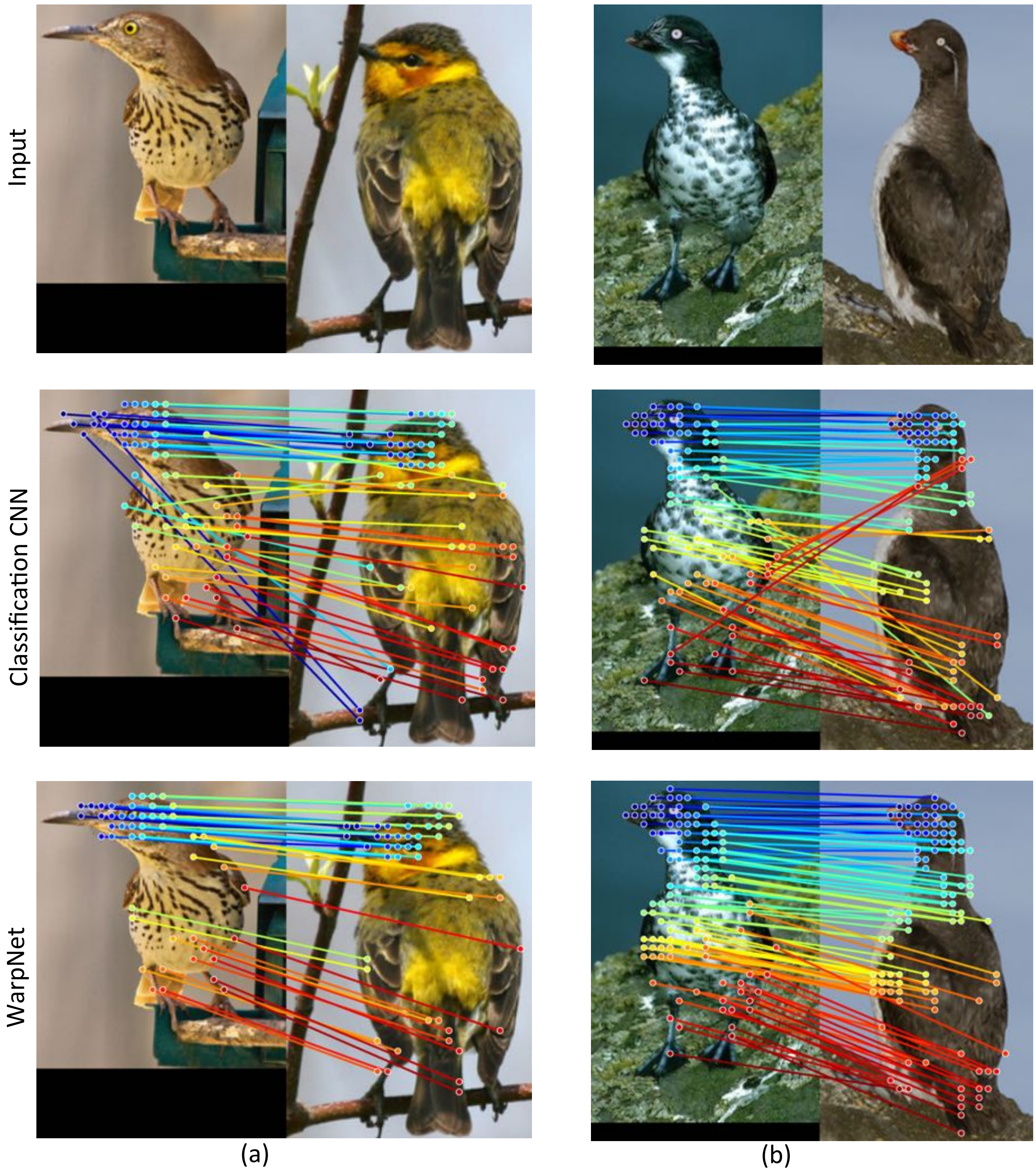


Figure 12: Failure cases for image pairs (row 1) that combine head articulation with similarity in the front and back of the bird body, using the features from a CNN trained on ILSVRC (row 2) and WarpNet (row 3) for matching.