DeepFlux for Skeletons in the Wild

Yukang Wang[1], Yongchao Xu[1*], Stavros Tsogkas[2,3,4†], Xiang Bai[1], Sven Dickinson[2,3,4†], Kaleem Siddiqi[5]

[1]Huazhong University of Science and Technology [2]University of Toronto
[3]Vector Institute for Artificial Intelligence [4]Samsung Toronto AI Research Center
[5]School of Computer Science and Centre for Intelligent Machines, McGill University

{wangyk,yongchaoxu,xbai}@hust.edu.cn, {tsogkas,sven}@cs.toronto.edu, siddiqi@cim.mcgill.ca

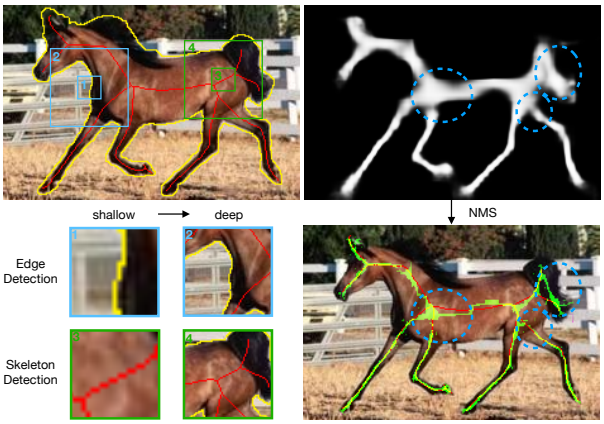
Abstract

Computing object skeletons in natural images is challenging, owing to large variations in object appearance and scale, and the complexity of handling background clutter. Many recent methods frame object skeleton detection as a binary pixel classification problem, which is similar in spirit to learning-based edge detection, as well as to semantic segmentation methods. In the present article, we depart from this strategy by training a CNN to predict a two-dimensional vector field, which maps each scene point to a candidate skeleton pixel, in the spirit of flux-based skeletonization algorithms. This "image context flux" representation has two major advantages over previous approaches. First, it explicitly encodes the relative position of skeletal pixels to semantically meaningful entities, such as the image points in their spatial context, and hence also the implied object boundaries. Second, since the skeleton detection context is a region-based vector field, it is better able to cope with object parts of large width. We evaluate the proposed method on three benchmark datasets for skeleton detection and two for symmetry detection, achieving consistently superior performance over state-of-the-art methods. The code is available at https://github.com/YukangWang/DeepFlux.

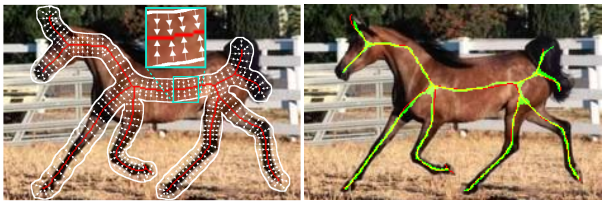
1. Introduction

The shape skeleton, or medial axis [3], is a structure-based object descriptor that reveals local symmetry as well as connectivity between object parts [31, 10]. Modeling objects via their axes of symmetry, and in particular, using skeletons, has a long history in computer vision. Skeletonization algorithms provide a concise and effective representation of deformable objects, while supporting many applications, including object recognition and retrieval [54, 14, 2, 45], pose estimation [15, 38, 48], hand gesture recognition [34], shape matching [41], scene text detection [52], and road detection in aerial scenes [44].

Early algorithms for computing skeletons directly from



(a) Previous CNN-based skeleton detections rely on NMS.



(b) Flux provides an alternative way for accurately detecting skeletons.

Figure 1. (a) Previous CNN-based methods treat skeleton detection as binary pixel classification, followed by non-maximum suppression (NMS). This can result in poor localization as well as poor connectedness. (b) The proposed DeepFlux method models skeleton context via a novel flux representation (left). The flux vector field encodes skeleton position in the context of the associated image pixels, and hence also the implied object boundaries. This allows one to associate skeletal pixels with sinks, where flux is absorbed, in the spirit of flux-based skeletonization methods [39]. Red: ground truth skeleton; Green: detected skeleton.

*Corresponding author

†Sven Dickinson and Stavros Tsogkas contributed to this article in their personal capacity as Professor and Postdoc, respectively, at the University of Toronto. The views expressed (or the conclusions reached) are their own and do not necessarily represent the views of Samsung Research America, Inc.

images [22, 25, 16, 50, 33, 51, 23] yield a gradient intensity map, driven by geometric constraints between skeletal pixels and edge fragments. Such methods cannot easily handle complex image data without prior information about object shape and location. Learning-based methods [21, 42, 47, 35, 44] have an improved ability for object skeleton detection in natural images, but such methods are still unable to cope with complex backgrounds or clutter.

The recent surge of work in convolutional neural networks (CNNs) has lead to vast improvements in the performance of object skeleton detection algorithms [37, 36, 19, 26, 53, 24]. These existing CNN-based methods usually derive from Holistically-Nested Edge Detection (HED) [49], and frame the problem as binary pixel classification. Most such approaches focus on designing an appropriate network and leveraging better multi-level features for capturing skeletons across a range of spatial scales.

Object skeleton computation using CNNs from natural images is inherently different from the problem of edge detection. As illustrated in Figure 1(a), edges associated with object boundaries can typically be detected locally, due to a local appearance change or a change in texture. Thus, the shallow convolutional layers, with accurate spatial information, can capture potential edge locations. Object skeletons, though, have to do with medial properties and high-level semantics. In particular, skeletons are situated at regions within object parts, where there is a local symmetry, since the medial axis bisects the object angle [40]. Capturing this purely from local image information (*e.g.*, the green box numbered 3 in Figure 1(a)) is not feasible, since this requires a larger spatial extent, in this case the width of the torso of the horse. Since shallow layers do not allow skeletal points to be captured, deeper layers of CNNs, with associated coarser features, are required. But this presents a confound – such coarse features may not provide accurate spatial localization of the object skeleton.

In this paper, we propose a novel notion of image context flux, to accurately detect object skeletons within a CNN framework. More precisely, we make use of skeleton context by using a two-dimensional vector field to capture a flux representation. For each skeleton context pixel, the flux is defined by the two-dimensional unit vector pointing to its nearest skeleton pixel. Within this flux representation, the object skeleton corresponds to pixels where the net inward flux is positive, following the motivation behind past flux-based methods for skeletonizing binary objects [39, 11]. We then develop a simple network to learn the image context flux, via a pixel-wise regression task in place of binary classification. Guided by the learned context flux encoding the relative location between context pixels and the skeleton, we can easily and accurately recover the object skeleton. In addition, the skeleton context provides a larger receptive field size for estimation, which is potentially helpful for detecting skeletons associated with larger spatial scales.

The main contributions of this paper are three-fold: 1) We propose a novel *context flux* to represent the object skeleton. This concept explicitly encodes the relationship between image pixels and their closest skeletal points. 2) Based on the context flux, we develop a method which we dub DeepFlux, that accurately and efficiently detects object skeletons in an image. 3) DeepFlux consistently outperforms state-of-the-art methods on five public benchmarks. To our knowledge this is the first application of flux concepts, which have been successfully used for skeletonization of binary objects, to the detection of object skeletons in natural images. It is also the first attempt at learning such flux-based representations directly from natural images.

The rest of this paper is organized as follows. We review related work in Section 2. We develop the DeepFlux method in Section 3 and carry out an extensive experimental evaluation in Section 4. We then conclude with a discussion of our results in Section 5.

2. Related Work

Object skeletonization has been widely studied in recent decades. In our review, we contrast traditional methods with those based on deep learning.

Traditional methods: Many early skeleton detection algorithms [22, 25, 16, 50, 33, 51, 23] are based on gradient intensity maps. In [39], the authors study the limiting average outward flux of the gradient of a Euclidean distance function to a 2D or 3D object boundary. The skeleton is associated with those locations where an energy principle is violated, where there is a net inward flux. Other researchers have constructed the skeleton by merging local skeleton segments with a learned segment-linking model. Levinshtein *et al.* [21] propose a method to work directly on images, which uses multi-scale super-pixels and a learned affinity between adjacent super-pixels to group proximal medial points. A graph-based clustering algorithm is then applied to form the complete skeleton. Lee *et al.* [42] improve the approach in [21] by using a deformable disc model, which can detect curved and tapered symmetric parts. A novel definition of an appearance medial axis transform (AMAT) has been proposed in [46], to detect symmetry in the wild in a purely bottom up, unsupervised fashion. In [17], the authors present an unconventional method based on joint co-skeletonization and co-segmentation.

In other literature [47, 35, 44], object skeleton detection is treated as a pixel-wise classification or regression problem. Tsogkas and Kokkinos [47] extract hand-designed features at each pixel and train a classifier for symmetry detection. They employ a multiple instance learning (MIL) framework to accommodate for the unknown scale and orientation of symmetry axes. Shen *et al.* [35] extend the ap-

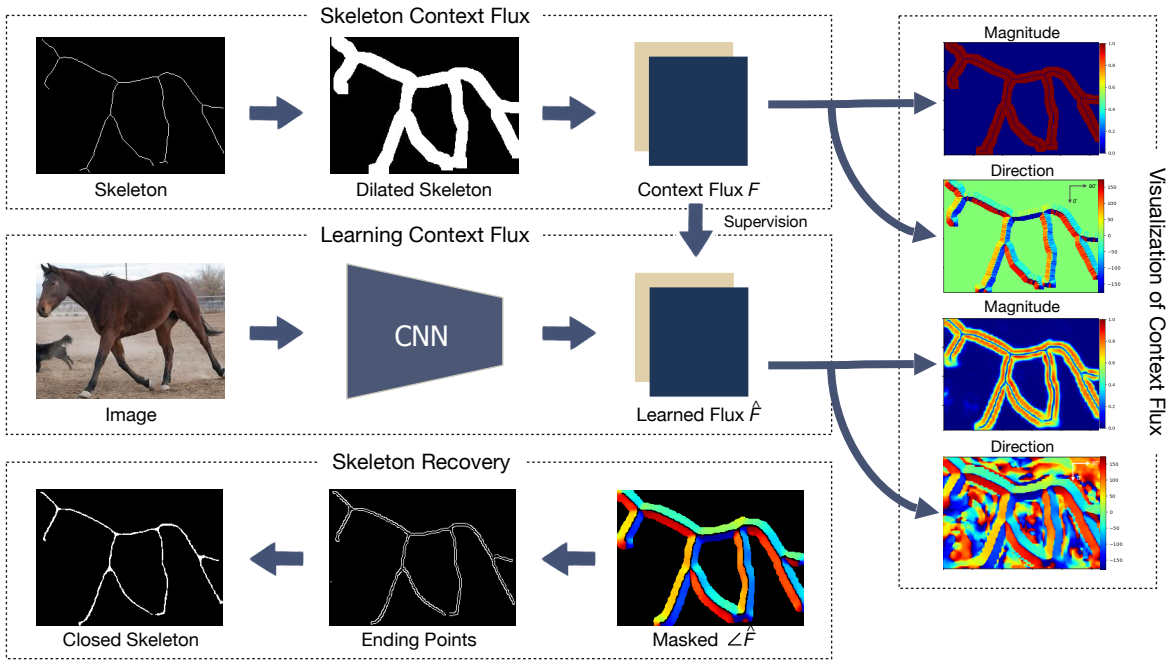


Figure 2. The DeepFlux pipeline. Given an input image, the network computes a two-dimensional vector field of skeleton context flux (visualizations of magnitude and direction on the right). The object skeleton is then recovered by localizing "ending points" where the net inward flux is high, followed by a morphological closing operation.

proach in [47] by training a group of MIL classifiers to capture the diversity of symmetry patterns. Sironi *et al.* [44] propose a regression-based approach to improve the accuracy of skeleton locations. They train regressors which learn the distances to the closest skeleton in scale-space and identify the skeleton by finding the local maxima.

Deep learning-based methods: With the popularization of CNNs, deep learning-based methods [37, 36, 19, 26, 53, 24] have had a tremendous impact on object skeleton detection. Shen *et al.* [37] fuse scale-associated deep side-outputs (FSDS) based on the architecture of HED [49]. Given that the skeleton of different scales can be captured in different stages, they supervise the side outputs with scale-associated ground-truth data. Shen *et al.* [36] then extend their original method by learning multi-task scale-associated deep side outputs (LMSDS). This leads to improved skeleton localization, scale prediction, and better overall performance. Ke *et al.* [19] present a side-output residual network (SRN), which leverages the output residual units to fit the errors between the ground-truth and the side-outputs. By cascading residual units in a deep-to-shallow manner, SRN can effectively detect the skeleton at different scales. Liu *et at.* [26] develop a two-stream network that combines image and segmentation cues to capture complementary information for skeleton localization. In [53], the authors introduce a hierarchical feature integration (Hi-Fi) mechanism. By hierarchically integrating multi-scale features with bidirectional guidance, high-level semantics and low-level details can benefit from each other. Liu *et al.* [24] propose a linear span network (LSN) that uses linear span units to increase the independence of convolutional features and the efficiency of feature integration.

Though the method we propose in the present paper benefits from CNN-based learning, it differs from the methods in [37, 36, 19, 26, 53, 24] in a fundamental way, due to its different learning objective. Instead of treating object skeleton detection in natural images as a binary classification problem, DeepFlux focuses on learning the context flux of skeletons, and as such includes more informative non-local cues, such as the relative position of skeleton points to image points in their vicinity, and thus also, implicitly, the associated object boundaries. A direct consequence of this powerful image context flux representation is that a simple post-processing step can recover the skeleton directly from the learned flux, avoiding inaccurate localizations of skeletal points caused by non-maximum suppression in previous deep learning methods. In addition, DeepFlux enlarges the spatial extent used by the CNN to detect the skeleton, through the use of skeleton context flux. This region-based flux representation allows our approach to capture larger object parts.

We note that the proposed DeepFlux is in spirit similar with the original notion of flux [39, 11] that is defined based on an object boundary, for skeletonization of 2D/3D binary objects. As such, DeepFlux inherits its mathematical properties including the unique mapping of skeletal points to

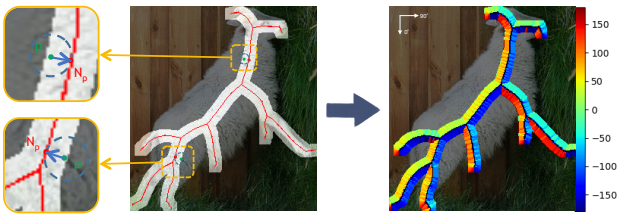


Figure 3. For each context (non-skeleton) pixel p in the dilated skeleton mask, we find its nearest skeleton pixel N_p. The flux $F(p)$ is defined as the two-dimensional unit vector that points away from p to N_p. For skeleton points, the flux is set to $(0, 0)$. On the right, we visualize the direction of the flux field.

boundary points. However, we are the first to extend this notion of flux to skeleton detection in natural images, by computing the flux on dilated skeletons for supervised learning. Our work is also related to the approaches in [1, 30, 5, 8] which learn the direction cues for edge detection and instance segmentation. In the present article, this direction information is encoded in the flux representation, and is implicitly learned for skeleton recovery.

3. Method

Many recent CNN-based skeleton detection approaches build on some variant of the HED architecture [49]. The combination of a powerful classifier (CNN) and the use of side outputs to extract and combine features at multiple scales has enabled these systems to accurately localize medial points of objects in natural images. However, while state-of-the-art skeleton detection systems are quite effective at extracting medial axes of elongated structures, they still struggle when reasoning about ligature areas. This is expected: contrary to the skeletal branches they connect, ligature areas exhibit much less structural regularity, making their exact localization ambiguous. As a result, most methods result in poor localization of ligature points, or poor connectedness between medial axes of object parts.

We propose to remedy this issue by casting skeleton detection as the problem of predicting a two-dimensional flux field from scene points to nearby skeleton points, within a fixed-size neighborhood. We then define skeleton points as the local flux minima, or, alternatively, as sinks "absorbing" flux from nearby points. We argue –and prove empirically in our experiments– that this approach leads to more robust localization and better connectivity between skeletal branches. We also argue that considering a small neighborhood around the true skeleton points is sufficient, consistent with past approaches to binary object skeletonization [11]. Whereas predicting the flux for the entire object would allow us to also infer the medial radius function, in this work we focus on improving medial point localization. The overall pipeline of the proposed method, aptly named *DeepFlux*, is depicted in Figure 2.

3.1. Skeleton context flux

We represent $F(x, y) = (F_x, F_y)$ as a two-channel map with continuous values corresponding to the x and y coordinates of the flux vector respectively. An intuitive visualization is shown in Figure 3. When skeleton detection is framed as a binary classification task, ground truth is a 1-pixel wide binary skeleton map; for our *regression* problem the ground truth must be modified appropriately.

We divide a binary skeleton map into three non-overlapping regions: 1) *skeleton context*, R_c, which is a the vicinity of the skeleton; 2) *skeleton pixels*, denoted by R_s; and 3) *background pixels*, R_b. In practice, R_c is obtained by dilating the binary skeleton map with a disk of radius r, and subtracting skeleton pixels R_s. Then, for each context pixel $p \in R_c$, we find its nearest (L_2 distance) skeleton pixel $N_p \in R_s$. A unit direction vector that points away from p to N_p is then computed as the flux on the context pixel p. This can be efficiently computed with the aid of a distance transform algorithm.* For the remaining pixels composed of R_s and R_b, we set the flux to $(0, 0)$. Formally, we have:

$$F(p) = \begin{cases} \overrightarrow{pN_p} / \left|\overrightarrow{pN_p}\right|, & p \in R_c \\ (0,0), & p \in R_s \cup R_b, \end{cases} \quad (1)$$

where $\left|\overrightarrow{pN_p}\right|$ denotes the length of the vector from p to N_p.

As a representation of the spatial context associated with each skeletal pixel, our proposed image context flux possesses a few distinct advantages when used to detect object skeletons in the wild. Unlike most learning approaches that predict skeleton probabilities individually for each pixel, our DeepFlux method leverages consistency between flux predictions within a neighborhood around each candidate pixel. Conversely, if the true skeleton location changes, the surrounding flux field will also change noticeably. A beneficial side-effect is that our method does not rely directly on the coarse responses produced by deeper CNN layers for localizing skeletons at larger scales, which further reduces localization errors. As we show in our experiments, these properties make our method more robust to the localization of skeleton points, especially around ligature regions, and less prone to gaps, discontinuities, and irregularities caused by local mispredictions. Finally, it is easy to accurately recover a binary object skeleton using the magnitude and direction of the predicted flux, as explained in Section 3.4.

3.2. Network architecture

The network for learning the skeleton context flux follows closely the fully convolutional architecture of [28], and

*In fact, in the context of skeletonization of binary objects [40], this flux vector would be in the direction opposite to that of the spoke vector from a skeletal pixel to its associated boundary pixel.

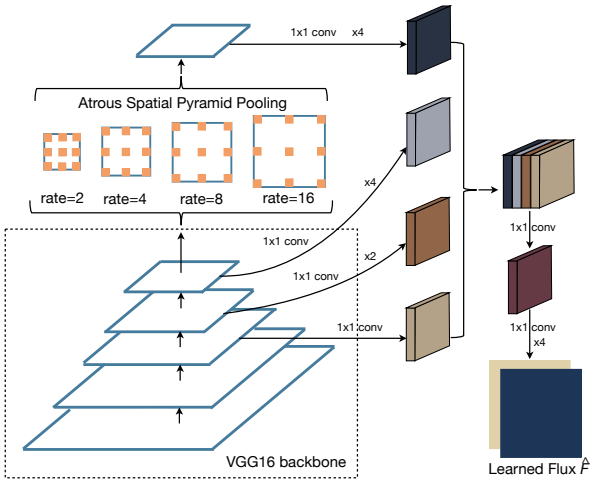


Figure 4. Network architecture. We adopt the pre-trained VGG16 [43] with the ASPP module [6] as the backbone network and with multi-level feature fusion via concatenation. The network is trained to predict the proposed context flux F, which is an image representing a two-dimensional vector field.

is shown in Figure 4. It consists of three modules: 1) a backbone network used to extract 3D feature maps; 2) an "atrous" spatial pyramid pooling (ASPP) module [6] to enlarge the receptive field while avoiding excessive downsampling; and 3) a multi-stage feature fusion module.

To ensure a fair comparison with previous work, we also adopt VGG16 [43] as the backbone network. As in [49], we discard the last pooling layer and the fully connected layers that follow. The use of the atrous module is motivated by the need for a wide receptive field: when extracting skeletons we have to guarantee that the receptive field of the network is wider than the largest medial radius of an object part in the input image. The receptive field of the VGG16 backbone is 196, which is not wide enough for large objects. Furthermore, it has been demonstrated in [29] that the effective receptive field only takes up a fraction of the full theoretical receptive field. Thus, we employ ASPP to capture multi-scale information. Specifically, four parallel atrous convolutional layers with 3×3 kernels but different atrous rates (2, 4, 8, 16) are added to the last layer of the backbone, followed by a concatenation along the channel dimension. In this way, we obtain feature maps with a theoretical receptive field size of 708 which we have found to be large enough for the images we have experimented on.

To construct a multi-scale representation of the input image, we fuse the feature maps from side outputs at conv3, conv4, conv5, and ASPP layers, after convolving them with a 1×1 kernel. Since feature maps at different levels have different spatial resolutions, we resize them all to the dimensions of conv3 before concatenating them. Prediction is then performed on the fused feature map, and then upsampled to the dimensions of the input image. For upsampling we use bilinear interpolation. The final output of the network is a 2-channel response map containing predictions of the x and y coordinates of the image content flux field $\hat{F}(p)$ for every pixel p in the image.

3.3. Training objective

We choose the L_2 loss function as our training objective. Due to a severe imbalance in the number of context and background pixels, we adopt a class-balancing strategy similar to the one in [49]. Our balanced loss function is

$$L = \sum_{p \in \Omega} w(p) * \left\| F(p) - \hat{F}(p) \right\|_2, \tag{2}$$

where Ω is the image domain, $\hat{F}(p)$ is the predicted flux, and $w(p)$ denotes the weight coefficient of pixel p. The weight $w(p)$ is calculated as follows:

$$w(p) = \begin{cases} \frac{|R_b|}{|R_c|+|R_b|+|R_s|}, & p \in R_c \cup R_s \\ \frac{|R_c|+|R_s|}{|R_c|+|R_b|+|R_s|}, & p \in R_b, \end{cases} \tag{3}$$

where $|R_c|$, $|R_b|$ and $|R_s|$ denote the number of context, background, and skeleton pixels, respectively.

3.4. From flux to skeleton points

We propose a simple post-processing procedure to recover an object skeleton from the predicted context flux. As described in Equation (1), pixels around the skeleton are labeled with unit two-dimensional vectors while the others are set to $(0, 0)$. Thus, thresholding the magnitude of the vector field reveals the context pixels while computing the flux direction reveals the location of context pixels relative to the skeleton. We refer the reader to Figure 2 for a visualization of the post-processing steps, listed in Algorithm 1.

Let $|\hat{F}|$ and $\angle\hat{F}$ be the magnitude and direction of the predicted context flux $\hat{F}$, respectively. For a given pixel p, $\angle\hat{F}(p)$ is binned into one of 8 directions, pointing to one of the 8 neighbors, denoted by $\mathcal{N}_{\angle\hat{F}(p)}(p)$. Having computed these two quantities, extracting the skeleton is straightforward: pixels close to the real object skeleton should have a high inward flux, due to a singularity in the vector field $\hat{F}$, as analyzed in [11]. These pixels are defined as "ending points". Finally, we apply a morphological dilation with a disk structuring element of radius k_1, followed by a morphological erosion with a disk of radius k_2, to group ending points together and produce the object skeleton.

4. Experiments

We conduct experiments on five well-known, challenging datasets, including three for skeleton detection (SK-LARGE [36], SK506 [37], WH-SYMMAX [35]) and two

Algorithm 1: Algorithm for skeleton recovery from learned context flux $\hat{F}$. $|\hat{F}|$: magnitude; $\angle\hat{F}$: direction; $\mathcal{N}_{\angle\hat{F}(p)}(p)$: neighbor of p at direction $\angle\hat{F}(p)$.

Input: Predicted context flux $\hat{F}$, threshold λ
Output: Binary skeleton map S

```
function Skeleton_recovery(F̂, λ)
    // initialization
    S ← False
    // find ending points near skeleton
    foreach p ∈ Ω do
        if |F̂(p)| > λ and |F̂(N_{∠F̂(p)}(p))| ≤ λ then
            S(p) ← True
    // apply morphological closing
    S ← ε_{k2}(δ_{k1}(S))
    return S
```

for local symmetry detection (SYM-PASCAL [19], SYMMAX300 [47]). We distinguish between the two tasks by associating skeletons with a foreground object, and local symmetry detection with any symmetric structure, be it a foreground object or background clutter.

4.1. Dataset and evaluation protocol

SK-LARGE [36] is a benchmark dataset for object skeleton detection, built on the MS COCO dataset [7]. It contains 1491 images, 746 for training and 745 for testing.

SK506 [37] (aka SK-SMALL), is an earlier version of SK-LARGE containing 300 train images and 206 test images.

WH-SYMMAX [35] contains 328 cropped images from the Weizmann Horse dataset [4], with skeleton annotations. It is split into 228 train images and 100 test images.

SYM-PASCAL [19] is derived from the PASCAL-VOC-2011 segmentation dataset [13] and targets object symmetry detection in the wild. It consists of 648 train images and 787 test images.

SYMMAX300 [47] is built on the Berkeley Segmentation Dataset (BSDS300) [32], which contains 200 train images and 100 test images. Both foreground and background symmetries are considered.

Evaluation protocol We use precision-recall (PR) curves and the F-measure metric to evaluate skeleton detection performance in our experiments. For methods that output a skeleton probability map, a standard non-maximal suppression (NMS) algorithm [12] is first applied and the thinned skeleton map is obtained. This map is then thresholded into a binary map and matched with the groundtruth skeleton map, allowing small localization errors. Since DeepFlux does not directly output skeleton probabilities, we use the inverse magnitude of predicted context flux on the recovered skeleton as a surrogate for a "skeleton confidence". Thresholding at different values gives rise to a PR curve and the optimal threshold is selected as the one producing the highest F-measure according to the formula $F = 2PR/(P+R)$. F-measure is commonly reported as a single scalar performance index.

4.2. Implementation details

Our implementation involves the following hyperparameters (values in parentheses denote the default values used in our experiments): the width of the skeleton context neighborhood $r = 7$; the threshold used to recover skeleton points from the predicted flux field, $\lambda = 0.4$; the sizes of the structuring elements involved in the morphological operations for skeleton recovery, $k_1 = 3$ and $k_2 = 4$.

For training, we adopt standard data augmentation strategies [37, 36, 53]. We resize training images to 3 different scales (0.8, 1, 1.2) and then rotate them to 4 angles (0°, 90°, 180°, 270°). Finally, we flip them with respect to different axes (up-down, left-right, no flip). The proposed network is initialized with the VGG16 model pretrained on ImageNet [9] and optimized using ADAM [20]. The learning rate is set to 10^{-4} for the first 100k iterations, then reduced to 10^{-5} for the remaining 40k iterations.

We use the Caffe [18] platform to train DeepFlux. All experiments are carried out on a workstation with an Intel Xeon 16-core CPU (3.5GHz), 64GB RAM, and a single Titan Xp GPU. Training on SK-LARGE using a batch size of 1 takes about 2 hours.

4.3. Results

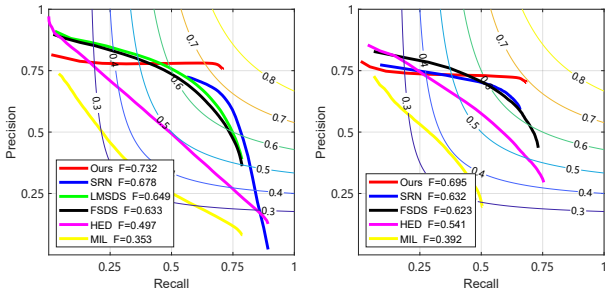
PR-curves for all methods are shown in Figure 5. DeepFlux performance excels particularly in the high-precision regime, where it clearly surpasses competing methods. This is indicative of the contribution of local context to more robust and accurate localization of skeleton points.

Table 1 lists the optimal F-measure score for all methods. DeepFlux consistently outperforms all other approaches using the VGG16 backbone [43]. Specifically, it improves over the recent Hi-Fi [53] by 0.8%, 1.4%, and 3.5% on SK-LARGE, SK506, and WH-SYMMAX, respectively, despite the fact that Hi-Fi uses stronger supervision during training (skeleton position *and* scale). DeepFlux also outperforms LSN [24], another recent method, by 6.4%, 6.2%, and 4.3% on SK-LARGE, SK506, and WH-SYMMAX, respectively.

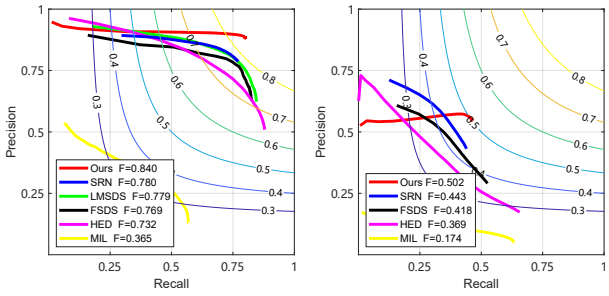
Similar results are observed for the symmetry detection task. DeepFlux significantly outperforms state-of-the-art methods on the SYM-PASCAL dataset, recording an improvement of 4.8% and 7.7% compared to Hi-Fi [53] and LSN [24], respectively. On SYMMAX300, DeepFlux also improves over LSN by 1.1%. Some qualitative results are

Methods	SK-LARGE	SK506	WH-SYMMAX	SYM-PASCAL	SYMMAX300
MIL [47]	0.353	0.392	0.365	0.174	0.362
HED [49]	0.497	0.541	0.732	0.369	0.427
RCF [27]	0.626	0.613	0.751	0.392	-
FSDS* [37]	0.633	0.623	0.769	0.418	0.467
LMSDS* [36]	0.649	0.621	0.779	-	-
SRN [19]	0.678	0.632	0.780	0.443	0.446
LSN [24]	0.668	0.633	0.797	0.425	0.480
Hi-Fi* [53]	0.724	0.681	0.805	0.454	-
DeepFlux (Ours)	**0.732**	**0.695**	**0.840**	**0.502**	**0.491**

Table 1. F-measure comparison. * indicates scale supervision was also used. Results for competing methods are from the respective papers.



(a) SK-LARGE (b) SK506



(c) WH-SYMMAX (d) SYM-PASCAL

Figure 5. PR curves on four datasets. DeepFlux offers high precision, especially in the high-recall regime.

shown in Figure 6, including failure cases.

4.4. Runtime analysis

We decompose runtime analysis into two stages: network inference and post-processing. Inference on the GPU using VGG16 takes on average 14 ms for a 300×200 image and the post-processing stage requires on average 3 ms on the CPU. As shown in Table 2, DeepFlux is as fast as competing methods while achieving superior performance.

4.5. Ablation study

We study the contribution of the two main modules (ASPP module and flux representation) to skeleton detection on SK-LARGE and SYM-PASCAL. We first remove the ASPP module and study the effect of the proposed context flux representation compared to a baseline model with

Method	F-measure	Runtime (in *sec*)
HED [49]	0.497	0.014
FSDS [37]	0.633	0.017
LMSDS [36]	0.649	0.019
LSN [24]	0.668	0.021
SRN [19]	0.678	0.016
Hi-Fi [53]	0.724	0.030
DeepFlux (ours)	0.732	0.017

Table 2. Runtime and performance on SK-LARGE. For DeeFlux we list the total inference (GPU) + post-processing (CPU) time.

the same architecture, but trained for binary classification. As shown in Table 3, employing a flux representation results in a 2.0% improvement on SK-LARGE and 4.9% on SYM-PASCAL. We then conduct experiments without using context flux, and study the effect of the increased receptive field offered by the ASPP module. The ASPP module alone leads to a 1.6% improvement on SK-LARGE and 1.7% on SYM-PASCAL. This demonstrates that the gains from ASPP and context flux are orthogonal; indeed, combining both improves the baseline model by $\sim 4\%$ on SK-LARGE and and $\sim 10\%$ on SYM-PASCAL.

We also study the effect of the size of the neighborhood within which context flux is defined. We conduct experiments with different radii, ranging from $r = 3$ to $r = 11$, on the SK-LARGE and SYM-PASCAL datasets. Best results are obtained for $r = 7$, and using smaller or larger values seems to slightly decrease performance. Our understanding is that a narrower context neighborhood provides less contextual information to predict the final skeleton map. On the other hand, using a wider neighborhood may increase the chance for mistakes in flux prediction around areas of severe discontinuities, such as the areas around boundaries of thin objects that are fully contained in the context neighborhood. The good news, however, is that DeepFlux is not sensitive to the value of r.

Finally, one may argue that simply using a dilated skeleton ground truth is sufficient to make a baseline model more robust in accurately localizing skeleton points. To examine if this is the case, we retrained our baseline model using bi-

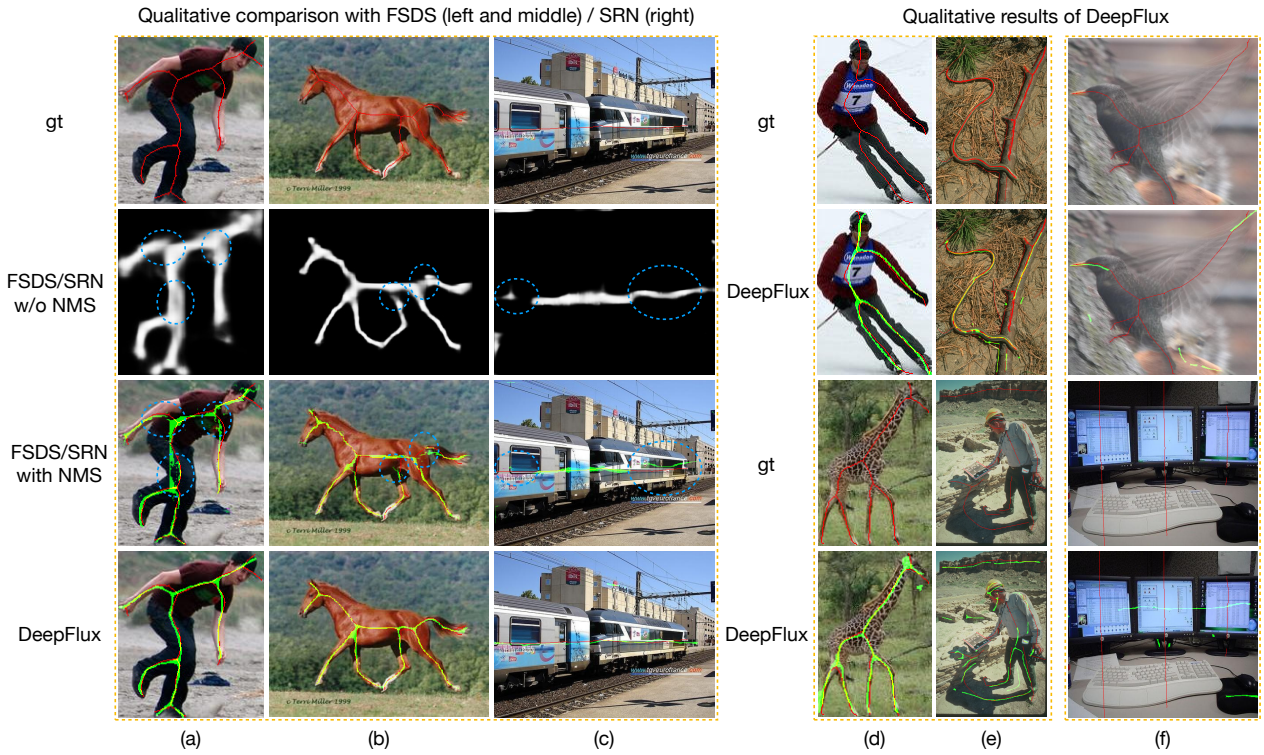


Figure 6. Qualitative results on SK-LARGE, WH-SYMMAX, and SYM-PASCAL (a-c), SK506 (d), SYMMAX300 (e), and two failure cases (f). Red: GT; Green: detected skeleton; Yellow: detected skeleton and GT overlap. DeepFlux fails to detect the skeleton on the bird body due the severe blurring. In the second failure example DeepFlux detects a symmetry axis not annotated in the ground truth.

Dataset	Context flux	ASPP	F-measure
SK-LARGE			0.696
		✓	0.712
	✓		0.716
	✓	✓	**0.732**
SYM-PASCAL			0.409
		✓	0.426
	✓		0.458
	✓	✓	**0.502**

Table 3. The effect of the context flux representation and the ASPP module on performance.

Dataset	$r = 3$	$r = 5$	$r = 7$	$r = 9$	$r = 11$
SK-LARGE	0.721	0.727	**0.732**	0.726	0.724
SYM-PASCAL	0.481	0.498	**0.502**	0.500	0.501

Table 4. Influence of the context size r on the F-measure.

nary cross-entropy on the same dilated ground truth we used for DeepFlux. Without context flux, performance drops to $F = 0.673$ (-6%) on SK-LARGE and to $F = 0.425$ (-8%) on SYM-PASCAL, validating the importance of our proposed representation for accurate localization.

5. Conclusion

We have proposed DeepFlux, a novel approach for accurate detection of object skeletons in the wild. Departing from the usual view of learning-based skeleton detection as a binary classification problem, we have recast it as the regression problem of predicting a 2D vector field of "context flux". We have developed a simple convolutional neural network to compute such a flux, followed by a simple post-processing scheme that can accurately recover object skeletons in $\sim 20ms$. Our approach steers clear of many limitations related to poor localization, commonly shared by previous methods, and particularly shines in handling ligature points, and skeletons at large scales.

Experimental results on five popular and challenging benchmarks demonstrate that DeepFlux systematically improves the state-of-the-art, both quantitatively and qualitatively. Furthermore, DeepFlux goes beyond object skeleton detection, and achieves state-of-the-art results in detecting generic symmetry in the wild. In the future, we would like to explore replacing the post-processing step used to recover the skeleton with an appropriate NN module, making the entire pipeline trainable in an end-to-end fashion.

Acknowledgement

This work was supported by NSFC 61703171 and 61573160, and in part by the NSF of Hubei Province of China under Grant 2018CFB199, to Dr. Yongchao Xu by the Young Elite Scientists Sponsorship Program by CAST.

References

[1] Min Bai and Raquel Urtasun. Deep watershed transform for instance segmentation. In *Proc. of CVPR*, pages 2858–2866, 2017. 4

[2] Xiang Bai, Xinggang Wang, Longin Jan Latecki, Wenyu Liu, and Zhuowen Tu. Active skeleton for non-rigid object detection. In *Proc. of ICCV*, pages 575–582, 2009. 1

[3] Harry Blum. Biological shape and visual science (part i). *Journal of theoretical Biology*, 38(2):205–287, 1973. 1

[4] Eran Borenstein and Shimon Ullman. Class-specific, top-down segmentation. In *Proc. of ECCV*, pages 109–122, 2002. 6

[5] Liang-Chieh Chen, Alexander Hermans, George Papandreou, Florian Schroff, Peng Wang, and Hartwig Adam. Masklab: Instance segmentation by refining object detection with semantic and direction features. In *Proc. of CVPR*, pages 4013–4022, 2018. 4

[6] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(4):834–848, 2018. 5

[7] Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco captions: Data collection and evaluation server. *CoRR*, abs/1504.00325, 2015. 6

[8] Hai Ci, Chunyu Wang, and Yizhou Wang. Video object segmentation by learning location-sensitive embeddings. In *Proc. of ECCV*, pages 501–516, 2018. 4

[9] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Fei-Fei Li. Imagenet: A large-scale hierarchical image database. In *Proc. of CVPR*, pages 248–255, 2009. 6

[10] Sven J Dickinson. *Object categorization: computer and human vision perspectives*. Cambridge University Press, 2009. 1

[11] Pavel Dimitrov, James N. Damon, and Kaleem Siddiqi. Flux invariants for shape. In *Proc. of CVPR*, 2003. 2, 3, 4, 5

[12] Piotr Dollár and C Lawrence Zitnick. Fast edge detection using structured forests. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(8):1558–1570, 2015. 6

[13] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 88(2):303–338, 2010. 6

[14] Pedro F Felzenszwalb and Daniel P Huttenlocher. Pictorial structures for object recognition. *International Journal of Computer Vision*, 61(1):55–79, 2005. 1

[15] Ross Girshick, Jamie Shotton, Pushmeet Kohli, Antonio Criminisi, and Andrew Fitzgibbon. Efficient regression of general-activity human poses from depth images. In *Proc. of ICCV*, pages 415–422, 2011. 1

[16] Jeong-Hun Jang and Ki-Sang Hong. A pseudo-distance map for the segmentation-free skeletonization of gray-scale images. In *Proc. of ICCV*, volume 2, pages 18–23, 2001. 2

[17] Koteswar Rao Jerripothula, Jianfei Cai, Jiangbo Lu, and Junsong Yuan. Object co-skeletonization with co-segmentation. In *Proc. of CVPR*, pages 3881–3889, 2017. 2

[18] Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell. Caffe: Convolutional architecture for fast feature embedding. In *Proc. of ACM-MM*, pages 675–678, 2014. 6

[19] Wei Ke, Jie Chen, Jianbin Jiao, Guoying Zhao, and Qixiang Ye. Srn: Side-output residual network for object symmetry detection in the wild. In *Proc. of CVPR*, pages 302–310, 2017. 2, 3, 6, 7

[20] D Kinga and J Ba Adam. A method for stochastic optimization. In *Proc. of ICLR*, volume 5, 2015. 6

[21] Alex Levinshtein, Cristian Sminchisescu, and Sven Dickinson. Multiscale symmetric part detection and grouping. *International Journal of Computer Vision*, 104(2):117–134, 2013. 2

[22] Tony Lindeberg. Edge detection and ridge detection with automatic scale selection. *International Journal of Computer Vision*, 30(2):117–156, 1998. 2

[23] Tony Lindeberg. Scale selection properties of generalized scale-space interest point detectors. *Journal of Mathematical Imaging and vision*, 46(2):177–210, 2013. 2

[24] Chang Liu, Wei Ke, Fei Qin, and Qixiang Ye. Linear span network for object skeleton detection. In *Proc. of ECCV*, pages 136–151, 2018. 2, 3, 6, 7

[25] Tyng-Luh Liu, Davi Geiger, and Alan L Yuille. Segmenting by seeking the symmetry axis. In *Proc. of ICPR*, volume 2, pages 994–998, 1998. 2

[26] Xiaolong Liu, Pengyuan Lyu, Xiang Bai, and Ming-Ming Cheng. Fusing image and segmentation cues for skeleton extraction in the wild. In *Proc. of ICCV Workshop on Detecting Symmetry in the Wild*, volume 6, page 8, 2017. 2, 3

[27] Yun Liu, Ming-Ming Cheng, Xiaowei Hu, Kai Wang, and Xiang Bai. Richer convolutional features for edge detection. In *Proc. of CVPR*, pages 5872–5881, 2017. 7

[28] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proc. of CVPR*, pages 3431–3440, 2015. 4

[29] Wenjie Luo, Yujia Li, Raquel Urtasun, and Richard Zemel. Understanding the effective receptive field in deep convolutional neural networks. In *Proc. of NIPS*, pages 4898–4906, 2016. 5

[30] Kevis-Kokitsi Maninis, Jordi Pont-Tuset, Pablo Arbeláez, and Luc Van Gool. Convolutional oriented boundaries: From image segmentation to high-level tasks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(4):819–833, 2018. 4

[31] David Marr and Herbert Keith Nishihara. Representation and recognition of the spatial organization of three-dimensional shapes. *Proceedings of the Royal Society of London B: Biological Sciences*, 200(1140):269–294, 1978. 1

[32] David Martin, Charless Fowlkes, Doron Tal, and Jitendra Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *Proc. of ICCV*, volume 2, pages 416–423, 2001. 6

[33] Alexandr Nedzved, Sergey Ablameyko, and Seiichi Uchida. Gray-scale thinning by using a pseudo-distance map. In *Proc. of ICPR*, 2006. 2

[34] Zhou Ren, Junsong Yuan, Jingjing Meng, and Zhengyou Zhang. Robust part-based hand gesture recognition using kinect sensor. *IEEE Transactions on Multimedia*, 15(5):1110–1120, 2013. 1

[35] Wei Shen, Xiang Bai, Zihao Hu, and Zhijiang Zhang. Multiple instance subspace learning via partial random projection tree for local reflection symmetry in natural images. *Pattern Recognition*, 52:306–316, 2016. 2, 5, 6

[36] Wei Shen, Kai Zhao, Yuan Jiang, Yan Wang, Xiang Bai, and Alan Yuille. Deepskeleton: Learning multi-task scale-associated deep side outputs for object skeleton extraction in natural images. *IEEE Transactions on Image Processing*, 26(11):5298–5311, 2017. 2, 3, 5, 6, 7

[37] Wei Shen, Kai Zhao, Yuan Jiang, Yan Wang, Zhijiang Zhang, and Xiang Bai. Object skeleton extraction in natural images by fusing scale-associated deep side outputs. In *Proc. of CVPR*, pages 222–230, 2016. 2, 3, 5, 6, 7

[38] Jamie Shotton, Andrew Fitzgibbon, Mat Cook, Toby Sharp, Mark Finocchio, Richard Moore, Alex Kipman, and Andrew Blake. Real-time human pose recognition in parts from single depth images. In *Proc. of CVPR*, pages 1297–1304, 2011. 1

[39] Kaleem Siddiqi, Sylvain Bouix, Allen Tannenbaum, and Steven W Zucker. Hamilton-jacobi skeletons. *International Journal of Computer Vision*, 48(3):215–231, 2002. 1, 2, 3

[40] Kaleem Siddiqi and Stephen M. Pizer. *Medial Representations: Mathematics, Algorithms and Applications*. Springer, 2008. 2, 4

[41] Kaleem Siddiqi, Ali Shokoufandeh, Sven J Dickinson, and Steven W Zucker. Shock graphs and shape matching. *International Journal of Computer Vision*, 35(1):13–32, 1999. 1

[42] Tom Sie Ho Lee, Sanja Fidler, and Sven Dickinson. Detecting curved symmetric parts using a deformable disc model. In *Proc. of ICCV*, pages 1753–1760, 2013. 2

[43] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *Proc. of ICLR*, 2015. 5, 6

[44] Amos Sironi, Vincent Lepetit, and Pascal Fua. Multiscale centerline detection by learning a scale-space distance transform. In *Proc. of CVPR*, pages 2697–2704, 2014. 1, 2, 3

[45] Nhon H Trinh and Benjamin B Kimia. Skeleton search: Category-specific object recognition and segmentation using a skeletal shape model. *International Journal of Computer Vision*, (2):215–240, 2011. 1

[46] Stavros Tsogkas and Sven Dickinson. AMAT: Medial axis transform for natural images. In *Proc. of ICCV*, pages 2727–2736, 2017. 2

[47] Stavros Tsogkas and Iasonas Kokkinos. Learning-based symmetry detection in natural images. In *Proc. of ECCV*, pages 41–54, 2012. 2, 3, 6, 7

[48] Shih-En Wei, Varun Ramakrishna, Takeo Kanade, and Yaser Sheikh. Convolutional pose machines. In *Proc. of CVPR*, pages 4724–4732, 2016. 1

[49] Saining Xie and Zhuowen Tu. Holistically-nested edge detection. In *Proc. of ICCV*, pages 1395–1403, 2015. 2, 3, 4, 5, 7

[50] Zeyun Yu and Chandrajit Bajaj. A segmentation-free approach for skeletonization of gray-scale images via anisotropic vector diffusion. In *Proc. of CVPR*, pages 415–420, 2004. 2

[51] Qiaoping Zhang and Isabelle Couloigner. Accurate centerline detection and line width estimation of thick lines using the radon transform. *IEEE Transactions on Image Processing*, 16(2):310–316, 2007. 2

[52] Zheng Zhang, Wei Shen, Cong Yao, and Xiang Bai. Symmetry-based text line detection in natural scenes. In *Proc. of CVPR*, pages 2558–2567, 2015. 1

[53] Kai Zhao, Wei Shen, Shanghua Gao, Dandan Li, and Ming-Ming Cheng. Hi-fi: Hierarchical feature integration for skeleton detection. In *Proc. of IJCAI*, pages 1191–1197, 2018. 2, 3, 6, 7

[54] Song Chun Zhu and Alan L Yuille. Forms: a flexible object recognition and modelling system. *International Journal of Computer Vision*, 20(3):187–212, 1996. 1