**U-Net: Convolutional Networks for Biomedical Image Segmentation**

Olaf Ronneberger, Philipp Fischer, and Thomas Brox

**Abstract.**

There is large consent that successful training of deep networks requires many thousand annotated training samples. In this paper, we present a network and training strategy that relies on the strong use of data augmentation to use the available annotated samples more efficiently. The architecture consists of a contracting path to capture context and a symmetric expanding path that enables precise localization. We show that such a network can be trained end-to-end from very few images and outperforms the prior best method (a sliding-window convolutional network) on the ISBI challenge for segmentation of neuronal structures in electron microscopic stacks. Using the same network trained on transmitted light microscopy images (phase contrast and DIC) we won the ISBI cell tracking challenge 2015 in these categories by a large margin. Moreover, the network is fast. Segmentation of a 512x512 image takes less than a second on a recent GPU. The full implementation (based on Caffe) and the trained networks are available at http://lmb.informatik.uni-freiburg.de/people/ronneber/u-net .

심층 네트워크의 성공적인 훈련에는 수천 개의 주석이 달린 훈련 샘플이 필요하다는 데 많은 동의가 있습니다. 이 백서에서는 사용 가능한 주석이 있는 샘플을 보다 효율적으로 사용하기 위해 데이터 증강의 강력한 사용에 의존하는 네트워크 및 훈련 전략을 제시합니다. 아키텍처는 컨텍스트를 캡처하기 위한 축소 경로와 정확한 위치 파악을 가능하게 하는 대칭 확장 경로로 구성됩니다. 우리는 그러한 네트워크가 아주 적은 수의 이미지에서 종단 간 훈련될 수 있고 전자 현미경 스택의 신경 구조 분할에 대한 ISBI 과제에 대한 이전 최고의 방법(슬라이딩 창 컨볼루션 네트워크)보다 성능이 우수함을 보여줍니다. 투과광 현미경 이미지(위상 대비 및 DIC)에 대해 훈련된 동일한 네트워크를 사용하여 이 범주에서 큰 차이로 ISBI 세포 추적 챌린지 2015에서 우승했습니다. 또한 네트워크가 빠릅니다. 512x512 이미지의 분할은 최근 GPU에서 1초도 채 걸리지 않습니다. 전체 구현(Caffe 기반) 및 훈련된 네트워크는 http://lmb.informatik.uni-freiburg.de/people /ronneber/u-net에서 사용할 수 있습니다.

**1 Introduction**

In the last two years, deep convolutional networks have outperformed the state of the art in many visual recognition tasks, e.g. [ 7 , 3]. While convolutional networks have already existed for a long time [ 8], their success was limited due to the size of the available training sets and the size of the considered networks. The breakthrough by Krizhevsky et al. [ 7] was due to supervised training of a large network with 8 layers and millions of parameters on the ImageNet dataset with 1 million training images. Since then, even larger and deeper networks have been trained [12].

지난 2년 동안 딥 컨볼루션 네트워크는 많은 시각적 인식 작업에서 최신 기술을 능가했습니다. [7,3]. 컨볼루션 네트워크는 이미 오래전부터 존재했지만[8] 사용 가능한 훈련 세트의 크기와 고려되는 네트워크의 크기로 인해 성공에 한계가 있었습니다. Krizhevsky et al. [7] 1백만 개의 훈련 이미지가 있는 ImageNet 데이터 세트에서 8개의 레이어와 수백만 개의 매개변수가 있는 대규모 네트워크의 감독된 훈련 때문입니다. 그 이후로 더 크고 더 깊은 네트워크가 훈련되었습니다

The typical use of convolutional networks is on classification tasks, where the output to an image is a single class label. However, in many visual tasks, especially in biomedical image processing, the desired output should include localization, i.e., a class label is supposed to be assigned to each pixel. Moreover, thousands of training images are usually beyond reach in biomedical tasks. Hence, Ciresan et al. [ 1] trained a network in a sliding-window setup to predict the class label of each pixel by providing a local region (patch) around that pixel as input. First, this network can localize. Secondly, the training data in terms of patches is much larger than the number of training images. The resulting network won the EM segmentation challenge at ISBI 2012 by a large margin.

컨볼루션 네트워크의 일반적인 사용은 분류 작업에 있으며, 여기서 이미지에 대한 출력은 단일 클래스 레이블입니다. 그러나 많은 시각적 작업, 특히 생의학 이미지 처리에서 원하는 출력에는 현지화가 포함되어야 합니다. 즉, 클래스 레이블이 각 픽셀에 할당되어야 합니다. 또한 수천 개의 훈련 이미지는 일반적으로 생물 의학 작업에서 도달할 수 없습니다. 따라서 Ciresan et al. [1]은 입력으로 해당 픽셀 주변의 로컬 영역(패치)을 제공하여 각 픽셀의 클래스 레이블을 예측하도록 슬라이딩 윈도우 설정에서 네트워크를 훈련했습니다. 첫째, 이 네트워크는 현지화할 수 있습니다. 둘째, 패치 측면에서 훈련 데이터가 훈련 이미지의 수보다 훨씬 큽니다. 그 결과 네트워크는 ISBI 2012에서 EM 세분화 챌린지에서 큰 차이로 우승했습니다.

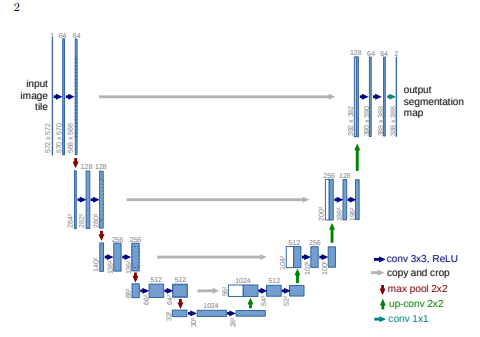


Fig. 1. U-net architecture (example for 32x32 pixels in the lowest resolution). Each blue box corresponds to a multi-channel feature map. The number of channels is denoted on top of the box. The x-y-size is provided at the lower left edge of the box. White boxes represent copied feature maps. The arrows denote the different operations.

그림 1. U-net 아키텍처(최저 해상도에서 32x32 픽셀의 예). 각 파란색 상자는 다중 채널 기능 맵에 해당합니다. 채널 수는 상자 상단에 표시됩니다. x-y-크기는 상자의 왼쪽 하단 가장자리에 제공됩니다. 흰색 상자는 복사된 기능 맵을 나타냅니다. 화살표는 다른 작업을 나타냅니다.

Obviously, the strategy in Ciresan et al. [1] has two drawbacks. First, it is quite slow because the network must be run separately for each patch, and there is a lot of redundancy due to overlapping patches. Secondly, there is a trade-off between localization accuracy and the use of context. Larger patches require more max-pooling layers that reduce the localization accuracy, while small patches allow the network to see only little context. More recent approaches [11,4] proposed a classifier output that takes into account the features from multiple layers. Good localization and the use of context are possible at the same time.

분명히, Ciresan et al. [1] 두 가지 단점이 있습니다. 첫째, 네트워크를 패치마다 별도로 실행해야 하기 때문에 상당히 느리고, 패치가 겹치기 때문에 중복성이 많다. 둘째, 현지화 정확도와 컨텍스트 사용 사이에는 절충점이 있습니다. 더 큰 패치는 지역화 정확도를 감소시키는 더 많은 최대 풀링 레이어를 필요로 하는 반면, 작은 패치는 네트워크가 약간의 컨텍스트만 볼 수 있도록 합니다. 보다 최근의 접근 방식은 여러 계층의 기능을 고려하는 분류기 출력을 제안했습니다. 좋은 현지화와 컨텍스트의 사용이 동시에 가능합니다.

In this paper, we build upon a more elegant architecture, the so-called “fully convolutional network” [9]. We modify and extend this architecture such that it works with very few training images and yields more precise segmentations; see Figure 1. The main idea in [9] is to supplement a usual contracting network by successive layers, where pooling operators are replaced by upsampling operators. Hence, these layers increase the resolution of the output. In order to localize, high resolution features from the contracting path are combined with the upsampled output. A successive convolution layer can then learn to assemble a more precise output based on this information.

이 백서에서 우리는 소위 "완전한 컨볼루션 네트워크"[9]라고 하는 보다 우아한 아키텍처를 기반으로 합니다. 우리는 이 아키텍처를 수정하고 확장하여 매우 적은 수의 훈련 이미지로 작동하고 보다 정확한 세분화를 생성합니다. 그림 1 참조. [9]의 주요 아이디어는 풀링 연산자가 업샘플링 연산자로 대체되는 연속적인 계층으로 일반적인 계약 네트워크를 보완하는 것입니다. 따라서 이러한 레이어는 출력의 해상도를 높입니다. 현지화를 위해 축소 경로의 고해상도 기능이 업샘플링된 출력과 결합됩니다. 그런 다음 연속적인 컨볼루션 계층은 이 정보를 기반으로 보다 정확한 출력을 조합하는 방법을 학습할 수 있습니다.

텍스트이(가) 표시된 사진

자동 생성된 설명

Fig. 2. Overlap-tile strategy for seamless segmentation of arbitrary large images (here segmentation of neuronal structures in EM stacks). Prediction of the segmentation in the yellow area, requires image data within the blue area as input. Missing input data is extrapolated by mirroring.

그림 2. 임의의 큰 이미지의 원활한 분할을 위한 중첩 타일 전략(여기서는 EM 스택의 신경 구조 분할). 노란색 영역의 분할 예측은 파란색 영역의 이미지 데이터가 입력으로 필요합니다. 누락된 입력 데이터는 미러링을 통해 외삽됩니다.

One important modification in our architecture is that in the upsampling part we have also a large number of feature channels, which allow the network to propagate context information to higher resolution layers. As a consequence, the expansive path is more or less symmetric to the contracting path, and yields a u-shaped architecture. The network does not have any fully connected layers and only uses the valid part of each convolution, i.e., the segmentation map only contains the pixels, for which the full context is available in the input image. This strategy allows the seamless segmentation of arbitrarily large images by an overlap-tile strategy (see Figure 2). To predict the pixels in the border region of the image, the missing context is extrapolated by mirroring the input image. This tiling strategy is important to apply the network to large images, since otherwise the resolution would be limited by the GPU memory.

우리 아키텍처의 한 가지 중요한 수정 사항은 업샘플링 부분에서 네트워크가 컨텍스트 정보를 고해상도 레이어로 전파할 수 있도록 하는 많은 수의 기능 채널도 있다는 것입니다. 결과적으로 확장 경로는 수축 경로와 다소 대칭적이며 U자형 아키텍처를 생성합니다. 네트워크에는 완전히 연결된 레이어가 없으며 각 컨볼루션의 유효한 부분만 사용합니다. 즉, 분할 맵에는 입력 이미지에서 전체 컨텍스트를 사용할 수 있는 픽셀만 포함됩니다. 이 전략을 사용하면 오버랩 타일 전략으로 임의의 큰 이미지를 매끄럽게 분할할 수 있습니다(그림 2 참조). 이미지의 경계 영역에 있는 픽셀을 예측하기 위해 누락된 컨텍스트는 입력 이미지를 미러링하여 외삽됩니다. 이 타일링 전략은 큰 이미지에 네트워크를 적용하는 데 중요합니다. 그렇지 않으면 GPU 메모리에 의해 해상도가 제한되기 때문입니다.

As for our tasks there is very little training data available, we use excessive data augmentation by applying elastic deformations to the available training images. This allows the network to learn invariance to such deformations, without the need to see these transformations in the annotated image corpus. This is particularly important in biomedical segmentation, since deformation used to be the most common variation in tissue and realistic deformations can be simulated efficiently. The value of data augmentation for learning invariance has been shown in Dosovitskiy et al. [2] in the scope of unsupervised feature learning.

우리 작업의 경우 사용 가능한 교육 데이터가 거의 없으므로 사용 가능한 교육 이미지에 탄성 변형을 적용하여 과도한 데이터 증대를 사용합니다. 이를 통해 네트워크는 주석이 달린 이미지 말뭉치에서 이러한 변형을 볼 필요 없이 이러한 변형에 대한 불변성을 학습할 수 있습니다. 조직의 가장 일반적인 변형이었던 변형과 현실적인 변형을 효율적으로 시뮬레이션할 수 있기 때문에 이것은 생물 의학 세분화에서 특히 중요합니다. 학습 불변성을 위한 데이터 증대의 가치는 Dosovitskiy et al. [2] 비지도 기능 학습의 범위에서.

Another challenge in many cell segmentation tasks is the separation of touching objects of the same class; see Figure 3. To this end, we propose the use of a weighted loss, where the separating background labels between touching cells obtain a large weight in the loss function.

많은 세포 분할 작업의 또 다른 과제는 동일한 클래스의 접촉 개체를 분리하는 것입니다. 그림 3을 참조하십시오. 이를 위해 가중 손실의 사용을 제안합니다. 여기서 터치 셀 사이의 분리 배경 레이블은 손실 함수에서 큰 가중치를 얻습니다.

The resulting network is applicable to various biomedical segmentation problems. In this paper, we show results on the segmentation of neuronal structures in EM stacks (an ongoing competition started at ISBI 2012), where we outperformed the network of Ciresan et al. [1]. Furthermore, we show results for cell segmentation in light microscopy images from the ISBI cell tracking challenge 2015. Here we won with a large margin on the two most challenging 2D transmitted light datasets.

결과 네트워크는 다양한 생물 의학 세분화 문제에 적용할 수 있습니다. 이 논문에서 우리는 Ciresan et al. [1]. 또한, 우리는 ISBI 세포 추적 챌린지 2015의 광학 현미경 이미지에서 세포 분할에 대한 결과를 보여줍니다. 여기에서 우리는 가장 어려운 두 가지 2D 투과광 데이터 세트에서 큰 차이로 승리했습니다.

**2 Network Architecture**

The network architecture is illustrated in Figure 1. It consists of a contracting path (left side) and an expansive path (right side). The contracting path follows the typical architecture of a convolutional network. It consists of the repeated application of two 3x3 convolutions (unpadded convolutions), each followed by a rectified linear unit (ReLU) and a 2x2 max pooling operation with stride 2 for downsampling. At each downsampling step we double the number of feature channels. Every step in the expansive path consists of an upsampling of the feature map followed by a 2x2 convolution (“up-convolution”) that halves the number of feature channels, a concatenation with the correspondingly cropped feature map from the contracting path, and two 3x3 convolutions, each followed by a ReLU. The cropping is necessary due to the loss of border pixels in every convolution. At the final layer a 1x1 convolution is used to map each 64- component feature vector to the desired number of classes. In total the network has 23 convolutional layers.

네트워크 아키텍처는 그림 1에 나와 있습니다. 이는 축소 경로(왼쪽)와 확장 경로(오른쪽)로 구성됩니다. 축소 경로는 컨볼루션 네트워크의 일반적인 아키텍처를 따릅니다. 이것은 두 개의 3x3 컨볼루션(패딩되지 않은 컨볼루션)의 반복적인 적용으로 구성되며, 각각은 정류 선형 단위(ReLU)와 다운샘플링을 위한 보폭 2를 사용하는 2x2 최대 풀링 작업이 뒤따릅니다. 각 다운샘플링 단계에서 기능 채널 수를 두 배로 늘립니다. 확장 경로의 모든 단계는 피쳐 맵의 업샘플링과 피쳐 채널 수를 절반으로 줄이는 2x2 컨볼루션("업 컨볼루션"), 축소 경로에서 잘린 피쳐 맵과의 연결 및 두 개의 3x3으로 구성됩니다. 각 회선 뒤에 ReLU가 옵니다. 모든 컨볼루션에서 경계 픽셀이 손실되기 때문에 자르기가 필요합니다. 최종 레이어에서 1x1 컨볼루션은 각 64개 구성 요소 특징 벡터를 원하는 클래스 수에 매핑하는 데 사용됩니다. 네트워크에는 총 23개의 컨볼루션 레이어가 있습니다.

To allow a seamless tiling of the output segmentation map (see Figure 2), it is important to select the input tile size such that all 2x2 max-pooling operations are applied to a layer with an even x- and y-size.

출력 분할 맵의 원활한 타일링을 허용하려면(그림 2 참조) 모든 2x2 최대 풀링 작업이 x 및 y 크기가 짝수인 레이어에 적용되도록 입력 타일 크기를 선택하는 것이 중요합니다.

**3 Training**

The input images and their corresponding segmentation maps are used to train the network with the stochastic gradient descent implementation of Caffe [6]. Due to the unpadded convolutions, the output image is smaller than the input by a constant border width. To minimize the overhead and make maximum use of the GPU memory, we favor large input tiles over a large batch size and hence reduce the batch to a single image. Accordingly we use a high momentum (0.99) such that a large number of the previously seen training samples determine the update in the current optimization step.

입력 이미지와 해당 분할 맵은 Caffe[6]의 확률적 경사하강법 구현으로 네트워크를 훈련하는 데 사용됩니다. 패딩되지 않은 컨볼루션으로 인해 출력 이미지는 일정한 테두리 너비만큼 입력보다 작습니다. 오버헤드를 최소화하고 GPU 메모리를 최대한 활용하기 위해 큰 배치 크기보다 큰 입력 타일을 선호하므로 배치를 단일 이미지로 줄입니다. 따라서 이전에 본 많은 훈련 샘플이 현재 최적화 단계의 업데이트를 결정하도록 높은 모멘텀(0.99)을 사용합니다.

The energy function is computed by a pixel-wise soft-max over the final feature map combined with the cross entropy loss function. The soft-max is defined as denotes the activation in feature channel k at the pixel position . K is the number of classes and pk(x) is the approximated maximum-function. I.e. pk(x) ≈ 1 for the k that has the maximum activation for all other k. The cross entropy then penalizes at each position the deviation of from 1 using

에너지 함수는 교차 엔트로피 손실 함수와 결합된 최종 피쳐 맵에 대해 픽셀 단위 소프트 맥스로 계산됩니다. soft-max는 여기서 는 인 픽셀 위치 에서 기능 채널 k의 활성화를 나타냅니다. K는 클래스 수이고 pk(x)는 근사 최대 함수입니다. 즉. 최대 활성화를 갖는 k에 대해 pk(x) ≈ 1이고 다른 모든 k에 대해 입니다. 그런 다음 교차 엔트로피는 다음을 사용하여 1에서 의 편차를 각 위치에서 페널티화합니다.



텍스트, 가위이(가) 표시된 사진

자동 생성된 설명

Fig. 3. HeLa cells on glass recorded with DIC (differential interference contrast) microscopy. (a) raw image. (b) overlay with ground truth segmentation. Different colors indicate different instances of the HeLa cells. (c) generated segmentation mask (white: foreground, black: background). (d) map with a pixel-wise loss weight to force the network to learn the border pixels.

그림 3. DIC(미분 간섭 대비) 현미경으로 기록된 유리 위의 HeLa 세포. (a) 원시 이미지. (b) ground truth 세분화로 오버레이. 다른 색상은 HeLa 세포의 다른 인스턴스를 나타냅니다. (c) 생성된 분할 마스크(흰색: 전경, 검은색: 배경). (d) 네트워크가 경계 픽셀을 학습하도록 하기 위해 픽셀 단위 손실 가중치로 매핑합니다.

where : Ω → {1, . . . , K} is the true label of each pixel and w : Ω → R is a weight map that we introduced to give some pixels more importance in the training.

여기서 : Ω → {1, . . . , K}는 각 픽셀의 실제 레이블이고 w : Ω → R은 학습에서 일부 픽셀에 더 많은 중요성을 부여하기 위해 도입한 가중치 맵입니다.

We pre-compute the weight map for each ground truth segmentation to compensate the different frequency of pixels from a certain class in the training data set, and to force the network to learn the small separation borders that we introduce between touching cells (See Figure 3c and d).

우리는 훈련 데이터 세트의 특정 클래스에서 픽셀의 다른 주파수를 보상하고 터치 셀 사이에 도입하는 작은 분리 경계를 네트워크가 학습하도록 강제하기 위해 각 정답 분할에 대한 가중치 맵을 미리 계산합니다(그림 3c 참조). 및 d).

The separation border is computed using morphological operations. The weight map is then computed as

분리 경계는 형태학적 연산을 사용하여 계산됩니다. 가중치 맵은 다음과 같이 계산됩니다.

텍스트이(가) 표시된 사진

자동 생성된 설명

where wc : Ω → R is the weight map to balance the class frequencies, d1 : Ω → R denotes the distance to the border of the nearest cell and d2 : Ω → R the distance to the border of the second nearest cell. In our experiments we set = 10 and σ ≈ 5 pixels.

여기서 wc : Ω → R은 클래스 주파수의 균형을 맞추기 위한 가중치 맵이고, d1 : Ω → R은 가장 가까운 셀 경계까지의 거리, d2 : Ω → R은 두 번째로 가까운 셀 경계까지의 거리입니다. 우리의 실험에서 우리는 w\_0 = 10 및 σ ≈ 5 픽셀을 설정했습니다.

In deep networks with many convolutional layers and different paths through the network, a good initialization of the weights is extremely important. Otherwise, parts of the network might give excessive activations, while other parts never contribute. Ideally the initial weights should be adapted such that each feature map in the network has approximately unit variance. For a network with our architecture (alternating convolution and ReLU layers) this can be achieved by drawing the initial weights from a Gaussian distribution with a standard deviation of , where N denotes the number of incoming nodes of one neuron [5]. E.g. for a 3x3 convolution and 64 feature channels in the previous layer N = 9 · 64 = 576

많은 컨볼루션 레이어와 네트워크를 통한 다양한 경로가 있는 심층 네트워크에서는 가중치를 잘 초기화하는 것이 매우 중요합니다. 그렇지 않으면 네트워크의 일부는 과도한 활성화를 제공하지만 다른 부분은 전혀 기여하지 않을 수 있습니다. 이상적으로는 네트워크의 각 기능 맵이 대략적인 단위 분산을 갖도록 초기 가중치를 조정해야 합니다. 우리 아키텍처(교대 컨볼루션 및 ReLU 레이어)가 있는 네트워크의 경우 표준 편차가 √(2/N)인 가우스 분포에서 초기 가중치를 가져와서 달성할 수 있습니다. 여기서 N은 한 뉴런의 들어오는 노드 수를 나타냅니다. 5]. 예를 들어 3x3 컨볼루션 및 이전 레이어의 64개 기능 채널의 경우 N = 9 · 64 = 576

**3.1 Data Augmentation**

Data augmentation is essential to teach the network the desired invariance and robustness properties, when only few training samples are available. In case of microscopical images we primarily need shift and rotation invariance as well as robustness to deformations and gray value variations. Especially random elastic deformations of the training samples seem to be the key concept to train a segmentation network with very few annotated images. We generate smooth deformations using random displacement vectors on a coarse 3 by 3 grid. The displacements are sampled from a Gaussian distribution with 10 pixels standard deviation. Per-pixel displacements are then computed using bicubic interpolation. Drop-out layers at the end of the contracting path perform further implicit data augmentation.

데이터 증대는 사용할 수 있는 훈련 샘플이 거의 없는 경우 네트워크에 원하는 불변성 및 견고성 속성을 가르치는 데 필수적입니다. 현미경 이미지의 경우 변형 및 회색 값 변동에 대한 견고성뿐만 아니라 이동 및 회전 불변성이 주로 필요합니다. 특히 훈련 샘플의 무작위 탄성 변형은 주석이 달린 이미지가 거의 없는 분할 네트워크를 훈련하는 핵심 개념인 것 같습니다. 거친 3 x 3 그리드에서 임의 변위 벡터를 사용하여 부드러운 변형을 생성합니다. 변위는 10픽셀 표준 편차를 갖는 가우스 분포에서 샘플링됩니다. 픽셀당 변위는 쌍삼차 보간을 사용하여 계산됩니다. 축소 경로 끝에 있는 드롭아웃 계층은 암시적 데이터 증대를 추가로 수행합니다.

**4 Experiments**

We demonstrate the application of the u-net to three different segmentation tasks. The first task is the segmentation of neuronal structures in electron microscopic recordings. An example of the data set and our obtained segmentation is displayed in Figure 2. We provide the full result as Supplementary Material. The data set is provided by the EM segmentation challenge [14] that was started at ISBI 2012 and is still open for new contributions. The training data is a set of 30 images (512x512 pixels) from serial section transmission electron microscopy of the Drosophila first instar larva ventral nerve cord (VNC). Each image comes with a corresponding fully annotated ground truth segmentation map for cells (white) and membranes (black). The test set is publicly available, but its segmentation maps are kept secret. An evaluation can be obtained by sending the predicted membrane probability map to the organizers. The evaluation is done by thresholding the map at 10 different levels and computation of the “warping error”, the “Rand error” and the “pixel error” [14].

우리는 u-net을 세 가지 다른 세분화 작업에 적용하는 방법을 보여줍니다. 첫 번째 작업은 전자 현미경 기록에서 신경 구조의 분할입니다. 데이터 세트의 예와 획득한 세분화가 그림 2에 표시됩니다. 전체 결과를 보충 자료로 제공합니다. 데이터 세트는 ISBI 2012에서 시작된 EM 세분화 챌린지[14]에 의해 제공되었으며 여전히 새로운 기여를 위해 열려 있습니다. 훈련 데이터는 초파리 1령 유충 복측 신경 코드(VNC)의 직렬 단면 투과 전자 현미경에서 얻은 30개의 이미지(512x512 픽셀) 세트입니다. 각 이미지에는 세포(흰색) 및 막(검은색)에 대해 완전히 주석이 달린 ground truth segmentation map이 함께 제공됩니다. 테스트 세트는 공개적으로 사용 가능하지만 세분화 맵은 비밀로 유지됩니다. 예측된 막 확률 맵을 주최자에게 전송하여 평가를 얻을 수 있습니다. 평가는 10가지 다른 수준에서 맵을 임계값으로 지정하고 "워핑 오류", "랜드 오류" 및 "픽셀 오류"를 계산하여 수행됩니다[14].

The u-net (averaged over 7 rotated versions of the input data) achieves without any further pre- or postprocessing a warping error of 0.0003529 (the new best score, see Table 1) and a rand-error of 0.0382.

u-net(입력 데이터의 7회전 버전에 대해 평균)은 0.0003529(새로운 최고 점수, 표 1 참조)의 왜곡 오류와 0.0382의 범위 오류를 추가 사전 또는 사후 처리 없이 달성합니다.

This is significantly better than the sliding-window convolutional network result by Ciresan et al. [1], whose best submission had a warping error of 0.000420 and a rand error of 0.0504. In terms of rand error the only better performing algorithms on this data set use highly data set specific post-processing methods1 applied to the probability map of Ciresan et al. [1]

이것은 Ciresan et al.의 슬라이딩 윈도우 컨볼루션 네트워크 결과보다 훨씬 낫습니다. [1], 최고의 제출은 0.000420의 뒤틀림 오류와 0.0504의 랜드 오류를 가졌습니다. 랜드 오류의 관점에서 이 데이터 세트에 대해 더 나은 성능을 보이는 알고리즘은 Ciresan et al.의 확률 맵에 적용된 고도로 데이터 세트별 후처리 방법1을 사용합니다. [1]

테이블이(가) 표시된 사진

자동 생성된 설명

Table 1. Ranking on the EM segmentation challenge [14] (march 6th, 2015), sorted by warping error.

표 1. EM 세분화 챌린지[14]의 순위(2015년 3월 6일), 뒤틀림 오류로 정렬됨.

텍스트이(가) 표시된 사진

자동 생성된 설명

Fig. 4. Result on the ISBI cell tracking challenge. (a) part of an input image of the “PhC-U373” data set. (b) Segmentation result (cyan mask) with manual ground truth (yellow border) (c) input image of the “DIC-HeLa” data set. (d) Segmentation result (random colored masks) with manual ground truth (yellow border).

그림 4. ISBI 셀 추적 챌린지 결과. (a) "PhC-U373" 데이터 세트의 입력 이미지 일부. (b) 수동 접지 진실(노란색 테두리)이 있는 분할 결과(청록색 마스크) (c) "DIC-HeLa" 데이터 세트의 입력 이미지. (d) 수동 ground truth(노란색 테두리)가 있는 분할 결과(무작위 색상 마스크).

텍스트이(가) 표시된 사진

자동 생성된 설명

Table 2. Segmentation results (IOU) on the ISBI cell tracking challenge 2015

표 2. ISBI 셀 트래킹 챌린지 2015에 대한 분할 결과(IOU)

We also applied the u-net to a cell segmentation task in light microscopic images. This segmenation task is part of the ISBI cell tracking challenge 2014 and 2015 [10,13]. The first data set “PhC-U373”2 contains Glioblastoma-astrocytoma U373 cells on a polyacrylimide substrate recorded by phase contrast microscopy (see Figure 4a,b and Supp. Material). It contains 35 partially annotated training images. Here we achieve an average IOU (“intersection over union”) of 92%, which is significantly better than the second best algorithm with 83% (see Table 2). The second data set “DIC-HeLa”3 are HeLa cells on a flat glass recorded by differential interference contrast (DIC) microscopy (see Figure 3, Figure 4c,d and Supp. Material). It contains 20 partially annotated training images. Here we achieve an average IOU of 77.5% which is significantly better than the second best algorithm with 46%.

우리는 또한 광학 현미경 이미지의 세포 분할 작업에 u-net을 적용했습니다. 이 분할 작업은 ISBI 셀 추적 챌린지 2014 및 2015의 일부입니다[10,13]. 첫 번째 데이터 세트 "PhC-U373"2에는 위상차 현미경으로 기록된 폴리아크릴이미드 기질에 교모세포종-성상세포종 U373 세포가 포함되어 있습니다(그림 4a,b 및 보조 자료 참조). 여기에는 부분적으로 주석이 달린 35개의 훈련 이미지가 포함되어 있습니다. 여기에서 우리는 92%의 평균 IOU("교집합을 통한 교차")를 달성했으며, 이는 83%로 두 번째로 좋은 알고리즘보다 훨씬 더 우수합니다(표 2 참조). 두 번째 데이터 세트 "DIC-HeLa"3는 차동 간섭 대비(DIC) 현미경으로 기록된 평면 유리의 HeLa 세포입니다(그림 3, 그림 4c, d 및 보조 재료 참조). 여기에는 부분적으로 주석이 달린 20개의 훈련 이미지가 포함되어 있습니다. 여기에서 우리는 77.5%의 평균 IOU를 달성했으며 이는 46%로 두 번째로 좋은 알고리즘보다 훨씬 더 우수합니다.

**5 Conclusion**

The u-net architecture achieves very good performance on very different biomedical segmentation applications. Thanks to data augmentation with elastic deformations, it only needs very few annotated images and has a very reasonable training time of only 10 hours on a NVidia Titan GPU (6 GB). We provide the full Caffe[6]-based implementation and the trained networks4 . We are sure that the u-net architecture can be applied easily to many more tasks.

u-net 아키텍처는 매우 다양한 생물의학 세분화 응용 프로그램에서 매우 우수한 성능을 달성합니다. 탄성 변형을 통한 데이터 증대 덕분에 주석이 달린 이미지가 거의 필요하지 않으며 NVidia Titan GPU(6GB)에서 단 10시간의 매우 합리적인 교육 시간을 갖습니다. 우리는 완전한 Caffe[6] 기반 구현과 훈련된 네트워크4를 제공합니다. 우리는 u-net 아키텍처가 더 많은 작업에 쉽게 적용될 수 있다고 확신합니다.

**Acknowlegements**

This study was supported by the Excellence Initiative of the German Federal and State governments (EXC 294) and by the BMBF (Fkz 0316185B).

이 연구는 독일 연방 및 주 정부의 Excellence Initiative(EXC 294)와 BMBF(Fkz 0316185B)의 지원을 받았습니다.