sensors

MDPI

Article

A Classification Method for the Cellular Images Based on Active Learning and Cross-Modal Transfer Learning

Caleb Vununu [1], Suk-Hwan Lee [2] and Ki-Ryong Kwon [1,*]

[1] Department of IT Convergence and Application Engineering, Pukyong National University, Busan 48513, Korea; exen.xmen@gmail.com

[2] Department of Computer Engineering, Dong-A University, Busan 49315, Korea; skylee@dau.ac.kr

* Correspondence: krkwon@pknu.ac.kr; Tel.: +82-51-629-6257

Abstract: In computer-aided diagnosis (CAD) systems, the automatic classification of the different types of the human epithelial type 2 (HEp-2) cells represents one of the critical steps in the diagnosis procedure of autoimmune diseases. Most of the methods prefer to tackle this task using the supervised learning paradigm. However, the necessity of having thousands of manually annotated examples constitutes a serious concern for the state-of-the-art HEp-2 cells classification methods. We present in this work a method that uses active learning in order to minimize the necessity of annotating the majority of the examples in the dataset. For this purpose, we use cross-modal transfer learning coupled with parallel deep residual networks. First, the parallel networks, which take simultaneously different wavelet coefficients as inputs, are trained in a fully supervised way by using a very small and already annotated dataset. Then, the trained networks are utilized on the targeted dataset, which is quite larger compared to the first one, using active learning techniques in order to only select the images that really need to be annotated among all the examples. The obtained results show that active learning, when mixed with an efficient transfer learning technique, can allow one to achieve a quite pleasant discrimination performance with only a few annotated examples in hands. This will help in building CAD systems by simplifying the burdensome task of labeling images while maintaining a similar performance with the state-of-the-art methods.

Keywords: HEp-2 cell images classification; computer-aided diagnosis; deep learning; active learning; transfer learning; pattern recognition

check for updates

Citation: Vununu, C.; Lee, S.-H.; Kwon, K.-R. A Classification Method for the Cellular Images Based on Active Learning and Cross-Modal Transfer Learning. *Sensors* **2021**, *21*, 1469. https://doi.org/10.3390/s21041469

Academic Editors: Yen-wei Chen and Lanfen Lin

Received: 18 January 2021
Accepted: 13 February 2021
Published: 20 February 2021

1. Introduction

The classification of the different types of the human epithelial type 2 (HEp-2) cells is one of the most important steps in the diagnosis procedure of autoimmune disease [1]. Performing this classification manually represents an arduous task and can cost a lot of time during the diagnosis process. Moreover, the manual analysis of the HEp-2 cell patterns poses a certain problem in terms of consistency of the diagnosis results, since the complexity of the images complicates the task for the pathologists [2]. This is the reason why the automatic discrimination of the different types of the HEp-2 cell images is more than necessary in order to help pathologists during the diagnosis procedure. Which makes the classification of these cells to be one of the important parts of the computer-aided diagnosis systems.

Different methods have been presented for this task in the literature. As a pattern recognition problem, the classification of the HEp-2 cells is usually tackled with a feature extraction part followed by a discrimination process. Feature extraction consists of extracting or selecting the information that is supposed to help differentiating the different cellular types. The second part of the process consists of utilizing the extracted features as the inputs of a discriminator (a classifier). Different hand-crafted features have been proposed for this purpose and many of them can be seen in the review by Foggia et al. [3].

Various descriptors like the discrete cosine transform [4,5], the scale-invariant feature transform [5,6], the local binary patterns [7–9] or many other different statistical features have been highlighted in the literature [10–15]. A multiclass support vector machine (SVM) is mostly used as the discriminator for these methods.

The automatic feature learning process afforded by deep learning has largely supplanted the use of these handcrafted features. In addition to the fact that the subjective choice of the features was a disadvantage for these methods in terms of consistency, their limitations in terms of the discrimination results explain why they have fallen out of use and been supplanted by the deep learning-based methods. In fact, currently, the quasi majority of the works in the literature utilize these methods in order to demonstrate their superiority over the conventional handcrafted features.

One of the pioneers works to adopt the convolutional neural network (CNN) for the HEp-2 cell classification was the method proposed by Foggia et al. [2] at the International Conference on Pattern Recognition (ICPR) HEp-2 cells classification contest in 2012. Since then, multiple works have proposed the use of CNN models in many different ways [16–21]. Among the most noticeable, Li et al. [22] have presented a customized CNN model, called the deep residual inception network (DRI-Net), which associates the residual connection from the ResNet [23] and the "Inception modules" utilized in the GoogleNet [24]. Additionally, among the most noticeable, Shen et al. [25] have used the ResNet approach but with a much deeper residual module with several cross connections between the layers. Their model was named the deep-cross residual network (DCR-Net) and they have tested a huge data augmentation process in order to boost the classification accuracy.

Interestingly, Majtner et al. [26] have proposed the use of generative adversarial networks [27], the deep convolutional generative adversarial networks (DCGAN) [28] specifically, in order to generate realistic artificial HEp-2 images. The goal was to augment the existing datasets with the artificial images generated by the DCGAN. Li et al. [29] have extended the idea presented in [23] by enlarging the convolutional kernels' size and adding more convolution operations with different dilations within the DRI modules. The short-cut connection is made outside the DRI module, i.e., the residual connection is made between the input and the final output of the module, while in the previous version [23], different residual connections were made between the layers inside the DRI module. It is necessary to mention that the researchers in this field, both for the handcrafted features-based and deep learning-based methods, as described above, prefer to adopt, quasi unanimously, the supervised learning paradigm in order to tackle the HEp-2 images classification.

Supervised learning necessitates the presence of labeled images. Consequently, even though the discrimination performance that we can obtain by using this methodology remains remarkable, the necessity of constructing labeled datasets that contain a considerable number of images represents a serious concern. In fact, deep learning-based methods require the presence of thousands of images and the process of labeling by hands these images can eventually represent a quite onerous task in the future, when we will have to create more expanded datasets, which can be a drawback for this methodology. In our previous works [30,31], we have explored the possibilities afforded by the unsupervised learning approach for this topic. In the present work, we adopt the supervised learning methodology but in a quite different manner compared to what is actually done in the HEp-2 cell images classification literature. The principal contribution of this work is the use of the techniques afforded by active learning in order to drastically minimize the need of the labeled images (in proportion of the total number of images) while maintaining a comparable performance with the actual state-of-the-art methods.

For this purpose, we propose to use active learning coupled with transfer learning. Transfer learning consists of using an already trained network for a new task. Fine-tuning the trained network consists of updating its parameters using the new dataset. This technique was used for the HEp-2 cells classification by Phan et al. [32], who utilized a model, the VGG-16 network [33], that was previously trained on the ImageNet dataset. Some others methods utilize a pretrained network only as a feature extractor. The high-level features extracted from the pretrained CNN model are then used in order to train a multiclass SVM. In the HEp-2 cell classification literature, this technique (transfer learning without fine-tuning) has been adopted in different ways by the works discussed next. Lu et al. [34] have used a pretrained VGG-16 network as a feature extractor while Nguyen et al. [35] proposed the use of an ensemble of networks. Another work using this idea was presented by Cascio et al. [36] with the use of the AlexNet [37] as the feature extractor. For all of them, the discrimination was performed by a SVM, or by both a SVM and a k-nearest neighbor classifier, as done in [36].

An interesting transfer learning approach, named cross-modal transfer learning, was proposed by Lei et al. [38]. Cross-modal transfer learning consists of updating the parameters of the pretrained network (fine-tuning) in two steps: first, by using a quite small dataset, then, second, by performing the update on the targeted dataset, which is supposed to be much larger and more complex than the first one. With the particularity that the two datasets have to be similar, i.e., they have to share the same feature domain. In fact, most of the pretrained CNN models were trained on the ImageNet dataset, which contains images that are far different with the HEp-2 cell images. The idea of cross-modal transfer learning is like performing a prefine-tuning (on the small dataset) before a final fine-tuning (on the targeted dataset) in order to smooth the parameters' updating process during the training. The authors in [38] have used ResNet-50 as the network to be fine-tuned. The small dataset utilized was the ICPR2012 dataset [2], and the targeted dataset was the ICPR2016 dataset [39], also known as the 13A dataset. Our method uses this cross-modal idea with a slight difference: we used the small dataset not in order to fine-tune an existing pretrained model, but in order to train our fully designed parallel deep networks.

Active learning regroups an ensemble of techniques whose aim is to minimize the data labeling burden while keeping the discrimination's efficiency unchanged. The idea is to use the network in a set of iterations in order to select, among the totality of the data, only the samples that really need to be annotated and used it for training the final model. Different methods can help to select these data, methods based on uncertainty sampling [40] or query-by-committee [41]. Some other works have used active learning with deep learning [42–45] and some others have proposed different techniques, such as neural-like structures based on geometrical transformation model [46], in order to ensure the possibility of obtaining satisfactory results even with a small number of training examples. We aim to demonstrate in this work that active learning techniques can also be applied in the HEp-2 cell images classification and allow one to minimize the need of labeled data while maintaining a comparable discrimination performance with the state-of-the-art deep learning-based methods that utilize the totality of the labeled images.

The contributions of the present work can be summarized as follows. First, we propose a dynamic learning method that uses two deep residual networks with the same structure in parallel in order to specifically tackle the intraclass variations and interclass similarities present in most of the HEp-2 cell images datasets. Two-dimensional (2D) discrete wavelet transform (DWT) is performed over the input images. The first network takes the approximation coefficients as inputs and the second network takes the sum of all the details coefficients as inputs. The learning of the two networks is done in parallel and their high-level features are fused at the end of the networks in order to mix the different information extracted from the two inputs.

Second, we use the idea of cross-modal transfer learning in order to boost the efficiency of the active learning. The two parallel deep residual networks are first trained with a small HEp-2 dataset, the SNPHEp-2 [5], which contains only around 2000 images, in order to build our pretrained model. The third and principal contribution is to apply this pretrained model on our targeted large-scale HEp-2 dataset, which will be presented in details in Section 3, and utilize the uncertainty sampling from active learning in order to only select, in that dataset, the most informative data.

Finally, we systemically investigate the effectiveness of the proposed method and our experimental results demonstrate that: first, the proposed parallel residual networks are effective for the discrimination of the HEp-2 cells; second, by using an efficient transfer learning methodology (similar to cross-modal transfer learning), active learning can effectively help to minimize the burden of labeling images by hands during the dataset creation by allowing us to select only a few number of informative data that really need to be annotated while maintaining a quite fair performance on a very large-scale dataset.

The remaining content of the paper is organized as follows. The next section (Section 2) presents in details each step of the proposed methodology. Section 3 presents the dataset, discusses the obtained results, and addresses a comparative study with the other supervised learning methods in the literature.

2. Proposed Method

The first step of our method is to create the parallel networks that we will use for the transfer and active learning. HEp-2 cell datasets have the particularity of denoting a significant heterogeneity. This is caused by the existence of mainly two different levels of fluorescence illumination (also denoted as intensity levels). Images shown in Figure 1 illustrate the disparities caused by the inhomogeneous fluorescence illumination. These disparities are the intraclass variations and the interclass similarities. Intraclass variations denote the variations within the same cellular type. Figure 1a shows a randomly selected positive intensity nucleolar cell image, while Figure 1b shows a randomly selected negative intensity nucleolar image. We can remark the strong disparities in terms of intensity between the two images even though they belong to the same class (intraclass variations). The same dissimilarities can be noticed between the two images depicted in Figure 1c,d, in case of the nuclear membrane cell type.

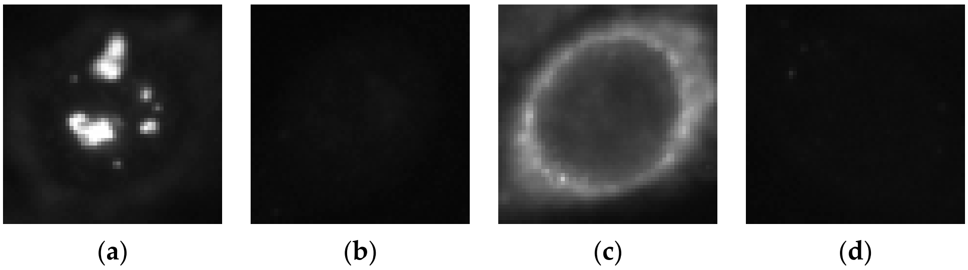


Figure 1. HEp-2 cellular images from the 13A dataset. (**a**) A positive intensity nucleolar cell; (**b**) a negative intensity nucleolar cell; (**c**) a positive intensity nuclear membrane cell and (**d**) a negative intensity nuclear membrane cell.

Interclass similarities, on the other hand, denote the similarities that exist between the different classes. In fact, the images shown in Figure 1b,d exhibit strong similarities in terms of intensity even though they belong to two different cellular types. This heterogeneity-related problem really adds complexity in the HEp-2 cell classification task. As a matter of fact, different methods have been proposed in order to specifically classify the different fluorescence intensity [47,48]. Furtherly, Nigam et al. [49] have proposed to perform an intensity-based classification prior to the cell classification itself in order to alleviate the heterogeneity during the cell type classification. Our proposed parallel deep residual networks try to tackle this heterogeneity-related problem in one step (unlike in [49]) and by performing cell type classification (unlike in [47,48]).

2.1. Parallel Deep Residual Networks

We propose to use the different wavelet coefficients from the 2D-DWT decomposition as the inputs of different networks in parallel. This idea was fully discussed and its effectiveness demonstrated in [50]. We upgraded the idea by alleviating the learning (training) process by reducing the total number of needed networks and, consequently, the total number of the parameters to handle. The 2D-DWT in the first level produces 4 different matrices of coefficients. The approximation coefficients, which represent the low-frequency information of the inputs, and the three different details coefficients, which represent the high frequency components of the input signal. The three details coefficients are the horizontal, vertical and diagonal details. Unlike in [50], where four different networks were utilized in parallel for all the four coefficients, we sum up all the details coefficients in order to incorporate all the high frequency components in one single channel. Thus, as illustrated in Figure 2, two networks are trained in parallel: the first network takes the approximation coefficients as the inputs, and the second network takes the sum of all the three details coefficients as inputs.

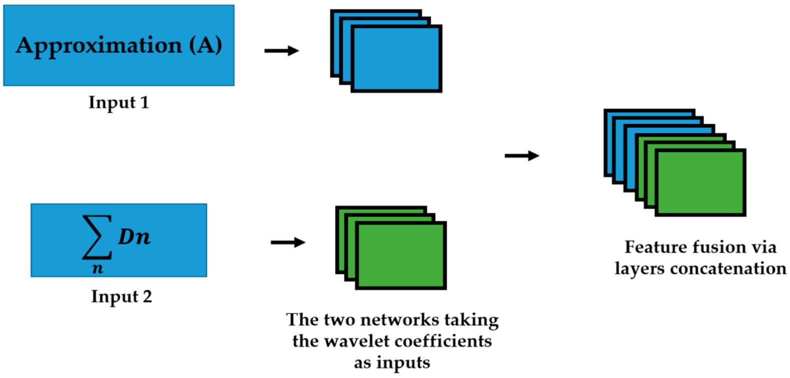

Approximation (A)
Input 1
$\sum_n Dn$
Input 2
The two networks taking the wavelet coefficients as inputs
Feature fusion via layers concatenation

Figure 2. The approximation (A) and the sum of details, represented by D, are given to two networks. A feature fusion is performed in the late layers. Note that *n* here is 3 and represents the three details components.

The approximation coefficients will bring a certain homogenization in terms of the intensity. This will drastically reduce the intraclass variations by forcing both the positive and negative intensity images to share a similar level of gray intensity. Images shown in Figure 3 illustrate the intensity-based homogenization produced by the approximation coefficients. Figure 3a shows a randomly selected positive intensity fine speckled cell image from the SNPHEp-2 dataset. Figure 3b shows its corresponding approximation coefficients (extracted from the first level of the 2D-DWT decomposition). Figure 3c shows a randomly selected negative intensity fine speckled cell image and Figure 3d shows its corresponding approximation coefficients. We can clearly remark the homogenization that occurred between the images in terms of the intensity of the gray level by comparing their approximation coefficients. This homogenization will drastically reduce the intraclass variations of the dataset.

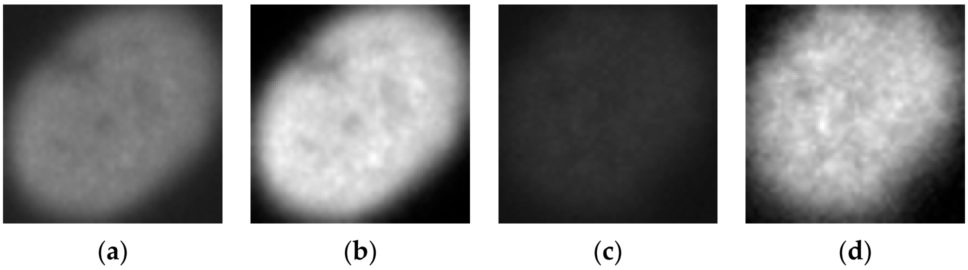


Figure 3. HEp-2 cellular images from the SNPHEp-2 dataset. (**a**) A positive intensity fine speckled cell; (**b**) its approximation coefficients; (**c**) a negative intensity fine speckled cell and (**d**) its approximation coefficients. Note the effective homogenization in terms of gray level intensity between the positive (**b**) and negative (**d**) images.

Secondly, the details coefficients will bring homogenization in terms of the geometrical shape of the cells. In fact, the details coefficients capture the high frequency components of the image, which means that all the gray variations inside the image can be highlighted.

These small changes in intensity indicate the shape and the boundaries of the cells. Images depicted in Figure 4 illustrate how the high-frequency components can help to expose the cellular shape and boundaries from the positive and negative intensity images.

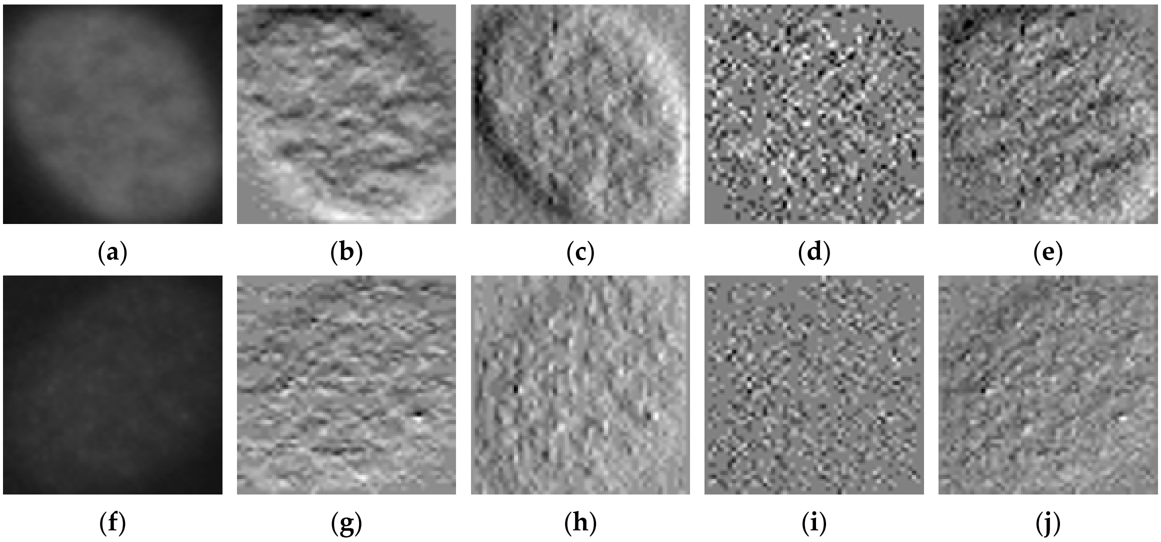
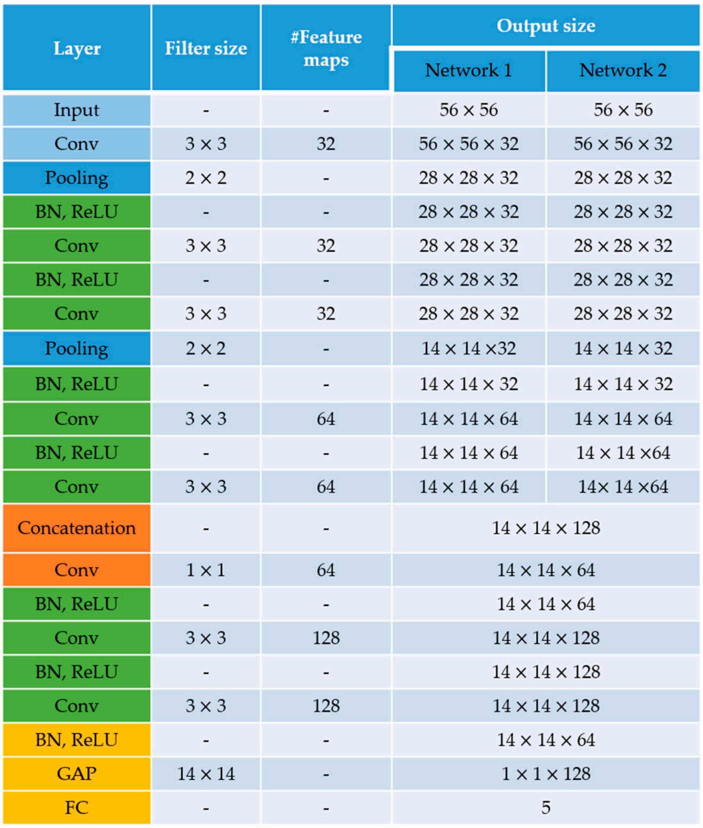


Figure 4. HEp-2 cellular images from the SNPHEp-2 dataset. (**a**) A positive intensity homogeneous cell and (**b**–**e**) its horizontal, vertical, diagonal details and their sum, respectively. (**f**) A negative intensity homogeneous cell and (**g**–**j**) its horizontal, vertical, diagonal details and their sum, respectively. The original images in (**a**,**f**) have a size of 112 × 112. Their respective detail coefficients in (**b**–**e**) and (**g**–**j**) are all downsized by half (56 × 56). All the images in the figure were identically resized for the purpose of visualization.

In Figure 4a, we show a positive intensity homogeneous cell. Images shown in Figure 4b–d represent its different details coefficients, the horizontal, vertical and diagonal details, respectively. The image shown in Figure 4e is the result of summing all the details. We can remark that the sum incorporates all the information from the three details coefficients. Similarly, in Figure 4f, we show a negative intensity homogeneous cell image. In Figure 4g–i, we show the three details coefficients and Figure 4j represents their sum. Note how the two sums (Figure 4e,j) highlight the homogeneous cell's shape, boundaries and internal gray variations. Since these three elements differ from a cellular type to another, we can expect two main contributions from the sum of details. First, they will bring a certain heterogeneity between the classes by forcing all the negative intensity images to exhibit typical characteristics of their cellular type (shape, boundaries and gray variations). This will contribute to the reduction of the interclass similarities.

Consequently, as the second contribution, they will bring a certain intraclass homogenization by forcing the positive and negative intensity images from the same class to exhibit similar patterns (shape, boundaries and gray variations), as demonstrated by Figure 4e,j. This will contribute to the reduction of the intraclass dissimilarities, reduction also achieved by the approximation coefficients, as previously discussed. The approximation and the sum of details will be used to feed the two residual networks in parallel.

Figure 5 shows the architecture of the residual networks. Network 1 takes the approximation coefficients while Network 2 takes the sum of the details, as explained above. There are five residual blocks in total. Each network has two residual blocks and another is used after the feature fusion from the two networks. Every residual block has two convolutional layers, two rectified linear unit (ReLU) layers and two batch normalization layers [51].



Layer	Filter size	#Feature maps	Output size	
			Network 1	Network 2
Input	-	-	56 × 56	56 × 56
Conv	3 × 3	32	56 × 56 × 32	56 × 56 × 32
Pooling	2 × 2	-	28 × 28 × 32	28 × 28 × 32
BN, ReLU	-	-	28 × 28 × 32	28 × 28 × 32
Conv	3 × 3	32	28 × 28 × 32	28 × 28 × 32
BN, ReLU	-	-	28 × 28 × 32	28 × 28 × 32
Conv	3 × 3	32	28 × 28 × 32	28 × 28 × 32
Pooling	2 × 2	-	14 × 14 ×32	14 × 14 × 32
BN, ReLU	-	-	14 × 14 × 32	14 × 14 × 32
Conv	3 × 3	64	14 × 14 × 64	14 × 14 × 64
BN, ReLU	-	-	14 × 14 × 64	14 × 14 ×64
Conv	3 × 3	64	14 × 14 × 64	14× 14 ×64
Concatenation	-	-	14 × 14 × 128	
Conv	1 × 1	64	14 × 14 × 64	
BN, ReLU	-	-	14 × 14 × 64	
Conv	3 × 3	128	14 × 14 × 128	
BN, ReLU	-	-	14 × 14 × 128	
Conv	3 × 3	128	14 × 14 × 128	
BN, ReLU	-	-	14 × 14 × 64	
GAP	14 × 14	-	1 × 1 × 128	
FC	-	-	5	

Figure 5. Architecture of the residual networks. The residual blocks are shown in green. Here, "Conv", "BN", "ReLU", "GAP" and "FC" denote, respectively, the convolutional layer, the batch normalization layer, the rectified linear unit (ReLU) layer, the global average pooling layer and the fully connected layer. "Pooling" denotes the maximum pooling layer and "Concatenation" denotes the layer that performs feature concatenation from the two networks.

Two main observations need to be made about the architecture in Figure 5. First, all the convolutional layers preserve the spatial dimensions of the input volume and only the pooling layers perform the spatial downsampling. Second, the feature fusion is performed by the 1 × 1 convolutional operation that directly follows the concatenation. After concatenating the layers from the two networks, we obtained a volume of size 14 × 14 × 128, which is then passed through the 1 × 1 convolutional layer whose purpose is precisely to mix (fusion) the information from the two networks. The output volume of the final residual block has the dimensions of 14 × 14 × 128. This volume is given to the global average pooling (GAP) layer in order to obtain the final one-dimensional feature vector of size 1 × 1 × 128.

The feature vector will be given to a softmax classifier [52] that uses the function defined as follows:

$$\sigma(z)_j = \frac{e^{z_j}}{\sum_{i=1}^{N} e^{z_i}}, \quad \text{for } j = 1, \ldots, N, \tag{1}$$

where N is the number of the classes and the values z are the inputs of the softmax function. The values $\sigma(z)_j$ are the outputs of the function and represent the probabilities of every class. The parallel networks learn by back-propagating the error [53] and using the cross-entropy error function [52] defined by

$$E = -\sum_{j=1}^{N} y_j \log\left[\sigma(z)_j\right], \tag{2}$$

where the values y_j denote the actual labels of the N classes for a given data and the values $\sigma(z)_j$ are the ones computed using Equation (1).

These parallel networks will be first trained with a small dataset, which contains only around one thousand training instances. After this initial training process, the networks will be utilized as the pretrained model in order to perform transfer learning coupled with active learning on the targeted dataset, which contains more than sixty thousand instances. As previously discussed in Section 1, instead of using networks that were pretrained on ImageNet, as done by most of the works that utilize transfer learning [32,34–36], we propose to use our own networks, which are pretrained purposely by using the HEp-2 images.

The advantage is that using a network that has previously seen similar images during transfer learning alleviates the parameters' update. Since the two datasets share the similar image domains, we expect them to share many general characteristics. The early layers from the networks, which learn low-level and non-specific features, can be fixed during the fine-tuning and only the late layers, which learn domain-specific features, can be updated (see Figure 6 for the illustration). This will smooth and ease the first step of our active learning scheme consisting of fine-tuning the pretrained model on a very small number of selected data.

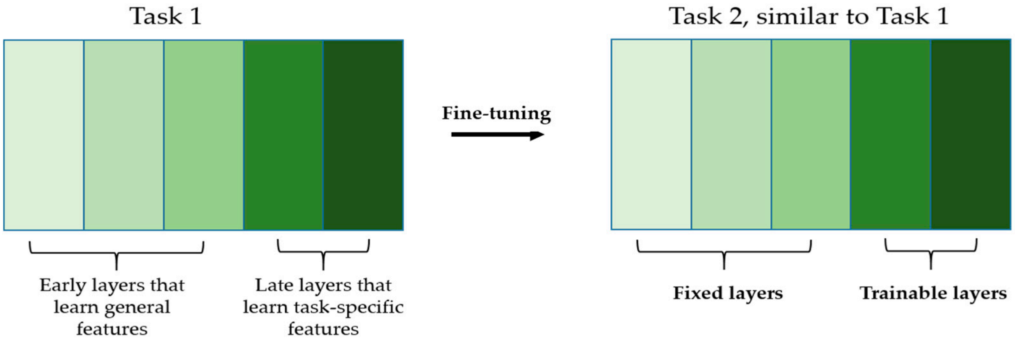

Task 1
Task 2, similar to Task 1
Fine-tuning
Early layers that learn general features
Late layers that learn task-specific features
Fixed layers
Trainable layers

Figure 6. Illustration of fine-tuning. Only the late layers will be trainable during the fine-tuning. The different shades of green indicate how the specificity of the features increases layer after layer. From left to right, we have the general features to the task-specific features.

2.2. *Active Learning Using the Pretrained Parallel Residual Networks*

Active learning aims to alleviate the labeling process by allowing one to select only the data that should be given to the learning model. The goal is to find which are the data that carry the most informative details that can help us to build the classifier. This selection can be done by finding the data for which the model is the most uncertain. This is referred as uncertainty sampling [40]. The idea is to find an uncertainty measure that can help to evaluate the confidence of the model and then use that measure in order to select only the data for which the model is the most uncertain. Select in order to annotate them. Instead of annotating (labeling by hands) all the available data, we can just annotate the ones for which our model is the most uncertain about.

Different uncertainty measures have been utilized in the literature. Among the most used, we have the entropy [54], which can measure the certainty level of a classifier by using the probabilities (scores) that are attributed to each class. For a given instance x, the entropy can be evaluated by the following equation:

$$entropy(x) = \sum_{j=1}^{N} p(y_j|x) \log p(y_j|x), \tag{3}$$

where the values $p(y_j|x)$ represent the classes' probabilities (or classes' scores) as outputted by the classifier for the data x and N is the number of classes. When the entropy is very high for a given data, it means that the classifier evaluates equally the different classes for that data. In other words, the classifier is uncertain about which class to assign to the data x.

Another way to measure the uncertainty of a model is to compute its amount of confidence for a given instance. This method is referred as least confidence-based sampling [40,55]. For a given data x, the confidence C is given by

$$C = \underset{j}{\mathrm{argmax}} p(y_j|x), \tag{4}$$

where the values $p(y_j|x)$ represent also the classes' probabilities. We know that for every single instance, the classifier outputs a vector containing the probabilities associated to each class, represented here by $p(y_j|x)$. The confidence denoted in Equation (4) finds the maximum score, i.e., it finds the class for which the model assigns the maximum probability value. Which gives us the amount of confidence of the model for every single data. The idea is to select the data for which the model is the least confident, i.e., the data for which the values C are the lowest. In practice, we can sort the data according to their confidence C from the smaller to the larger, and then prioritize the annotation in that manner. Note that entropy also can be thought as the amount of confidence: when the entropy is high, the confidence is low, and vice versa.

Another method, called the margin sampling [56], consists of computing the difference between the highest and the second highest scores in the vector of probabilities. In this case, the confidence C is given by

$$C = p_1(y_j|x) - p_2(y_j|x), \tag{5}$$

where $p_1(y_j|x)$, and $p_2(y_j|x)$ are the highest and the second highest probabilities, respectively. Similar with the previous case, the data with the lower C values are prioritized for the labeling. The difference computed in Equation (5) enquires us about the confidence of the model. If its value is high, it means that one of the class has a much bigger probability compared to the other and, on the contrary, when its value is small, it means that the model evaluates equally the two classes, which means that the model is uncertain.

The least confidence and margin sampling methods work similarly and, unlike the entropy, are both less suitable for the multiclass classification. In fact, in Equation (5), only two classes are taken into account, while the entropy utilizes all the classes' probabilities. In our case, we will adopt the entropy in order to evaluate the uncertainty of the deep parallel residual networks. Other uncertainty sampling methods can be found in [57–59]. For clarity, we summarize the different steps of the proposed active learning scheme in Table 1. Figure 7 illustrates these different steps.

This process is repeated as much as possible and stopped until we reach our limitations in terms of labeling. Note that this iterative process can be continued until the totality of the data are labeled. However, in our work, we imposed to ourselves a limit in terms of the possibility of labeling. In fact, the goal of this work was to demonstrate that active learning-based labeling could really help to limit the need of labeled data while maintaining a fair performance. In our experiments, we explored different hypotheses concerning the limitations that we had in terms of labeling. For example, if we suppose that we can only label 10% of the 64,000 available data, we stop the process when we reach 6400 annotated data and evaluate the networks over the testing data. All the details concerning the parameters of the networks, the values k and m, and the datasets are discussed in the next section.

Table 1. The different steps of our active learning scheme.

Step No.	Actions	Comments
1	We train our networks using the small dataset.	The small dataset used here is the SNPHEp-2 dataset, which contains around 1000 images for training.
2	Using the targeted dataset, we select randomly and label k samples. We fine-tune the networks by using these k samples as the training data. As described in Section 2.1, the early layers remain fixed and we only update the late layers.	Note that by choosing the number k, we select randomly (as opposed to select by using active learning) the data to label. In fact, we want this number k to be as small as possible, in order to not complicate the labeling process. This is made possible by the pretraining made in step 1 using the small dataset.
3	We use the fine-tuned networks over all the remaining data in order to get their probability scores. We compute the confidence (entropy) using Equation (3) for each data.	Equations (4) and (5) can also be used to estimate the confidence.
4	We rank the data according to their confidence, from the lowest to the highest.	Note that in Figure 7, we show the data with the lowest confidence (highest entropy) in the bottom for the illustration purpose.
5	We select the first m data in the ranking in step 4 and annotate them. These are the data for which the networks are the most uncertain about.	The number m is chosen according to the limitations that we have in terms of manual labeling.
6	The newly annotated data in step 5 are mixed with the k data that were previously labeled in step 2 in order to create the newly annotated set.	The newly annotated dataset contains now $k + m$ data.
7	We fine-tune again the networks using this newly annotated dataset	After this step, we get back to step 3 (use the newly fine-tuned networks to compute the scores and the confidence).

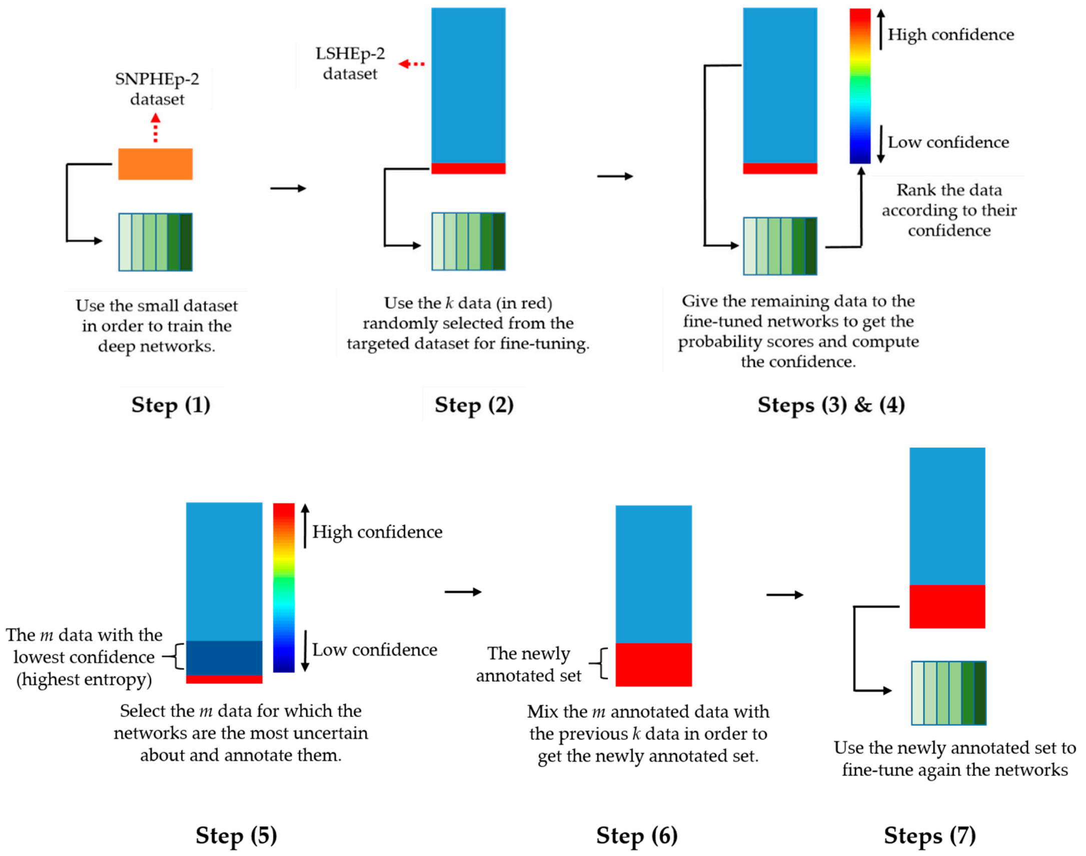

SNPHEp-2 dataset
Use the small dataset in order to train the deep networks.
Step (1)
LSHEp-2 dataset
Use the k data (in red) randomly selected from the targeted dataset for fine-tuning.
Step (2)
High confidence
Low confidence
Rank the data according to their confidence
Give the remaining data to the fine-tuned networks to get the probability scores and compute the confidence.
Steps (3) & (4)
High confidence
Low confidence
The m data with the lowest confidence (highest entropy)
Select the m data for which the networks are the most uncertain about and annotate them.
Step (5)
The newly annotated set
Mix the m annotated data with the previous k data in order to get the newly annotated set.
Step (6)
Use the newly annotated set to fine-tune again the networks
Steps (7)

Figure 7. Illustration of the different steps of the proposed active learning scheme. LSHEp-2 stands for the large-scale HEp-2 dataset.

3. Results

3.1. Datasets, Experimental Setup and Initial Learning Results

The deep residual networks were first trained using a relatively small dataset, as mentioned before. We adopted the SNPHEp-2 dataset here for this initial learning process. This dataset was presented by Wiliem et al. [5] and comprises five classes, which explains why the classification layer of the networks shown in Figure 5 had five neurons. The five classes (cell types) are the homogeneous, the coarse speckled, the fine speckled, the nucleolar and the centromere cells. The dataset contains 1884 data, divided into two sets: 905 images for training and 979 others for testing. Figure 8a–e shows one sample per class from this dataset.

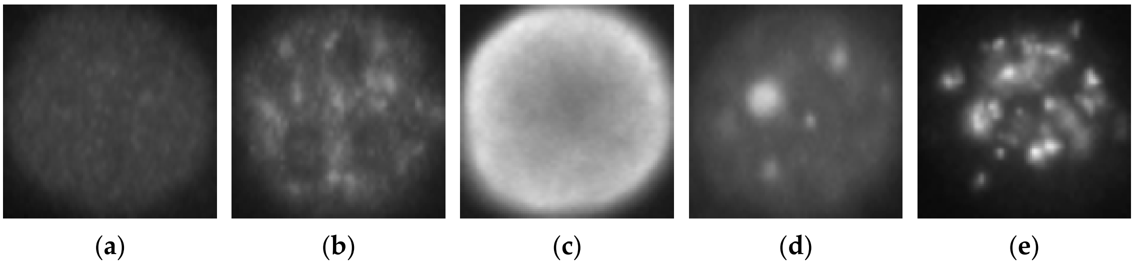


Figure 8. Example images from the SNPHEp-2 dataset. (**a**) The homogeneous, (**b**) the coarse speckled, (**c**) the fine speckled, (**d**) the nucleolar and (**e**) the centromere cells. The original size of the images is 112 × 112.

Instead of using the ICPR2012 for the initial learning as done in [38], the choice of using the SNPHEp-2 dataset was justified by the similarities between this dataset and our targeted dataset. The targeted dataset is the large-scale HEp-2 (LSHEp-2) dataset, introduced by Qi et al. [60]. This dataset contains far more images (63,445) than the 13A dataset (13,596 images). The reason why we adopted the LSHEp-2 dataset is that we aimed to test the effectiveness of the proposed active learning scheme on a really big dataset for which labeling can really be burdensome. Furthermore, this dataset is more complex in terms of intraclass variations and heterogeneity compared to the others. The description of the LSHEp-2 dataset can be found in details in [60] and it can be downloaded at http://qixianbiao.github.io/HEp2Cell/ (accessed on 16 January 2021). Similar to the 13A dataset, it contains six classes: homogeneous, speckled, nucleolar, centromere, nuclear membrane and Golgi cells. Some examples of this dataset are shown in Figure 9.

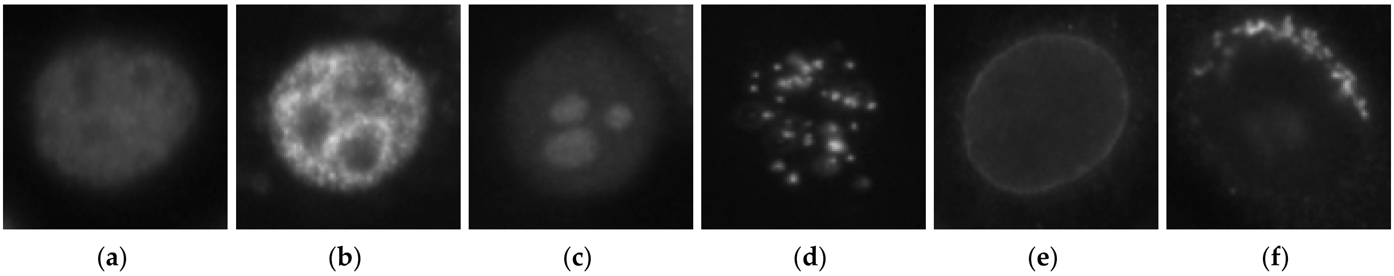
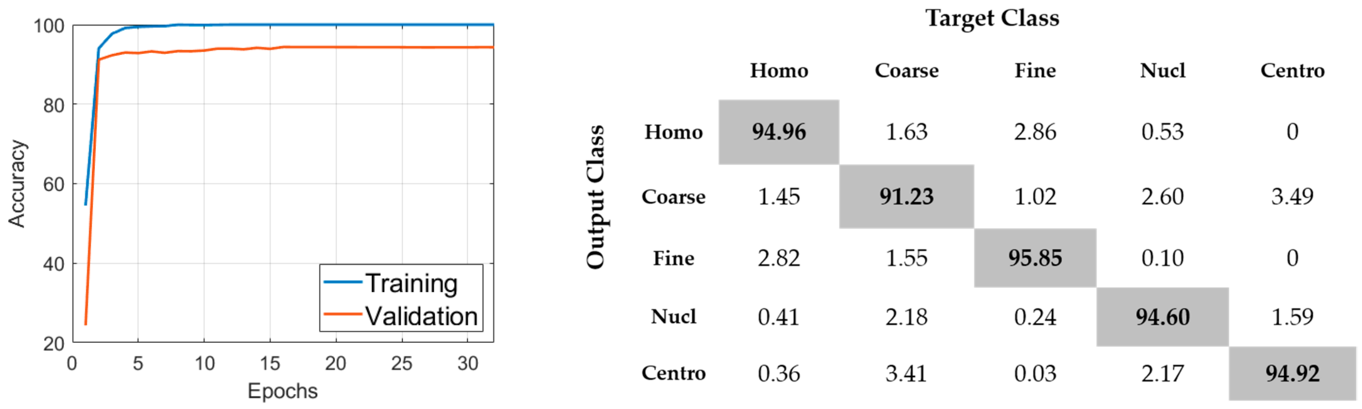


Figure 9. Example images from the large-scale HEp-2 dataset. (**a**) The homogeneous, (**b**) the speckled, (**c**) the nucleolar, (**d**) the centromere, (**e**) the nuclear membrane cells and (**f**) the Golgi cells.

By comparing the images in Figures 8 and 9, we can remark that the two datasets share many similarities. We can expect that our networks will learn the general features shared by these two sets of images, which will allow us to only update the task-specific layers located at the end of the networks. In fact, two big changes can be remarked between the SNPHEp-2 and the LSHEp-2 datasets: firstly, the two speckled (fine and coarse) cells from the first were mixed to form only one cell type, the speckled cells, in the second. Secondly, the Golgi cells are absent from the SNPHEp-2. During the transfer learning process, we will remove the last layer containing five neurons and replace it by another one containing six. All the experiments were performed using TensorFlow on a computer with a Core i7 3.40 GHz processor, 8 GB of RAM and a NVIDIA GeForce GTX 1080 Ti GPU.

For the initial learning process, the hyperparameters were selected via cross-validation using the five different validation folds of the SNPHEp-2 dataset. The original images had different sizes (average around 90 × 90) and were all upsized using bicubic interpolation to 112 × 112 in order to fit into our designed architecture. Note that after the DWT decomposition, the coefficients at the first level have all the size of 56 × 56. In order to maximize the learning capacity of the networks, data augmentation was applied over the SNPHEp-2 dataset. It consists of cell rotation, with a step of 18° in a quadrant of 360°, as proposed in [16,25]. This rotation increases the original training set by a factor of 20.

The learning rate is set to be 0.001 and training is terminated when the validation loss did not surpass the reached minimum 5 times in a row. For the initial learning (with data augmentation), 32 epochs were necessary to terminate the training process (see Figure 10a). The classification results of this initial learning are shown in the confusion matrix depicted in Figure 10b. The accuracy over the validation set, as we can see in Figure 10, was about 94%. We recall here that the purpose of this initial learning is just to generate a pretrained model that will be used later for the transfer learning. For further details about the effectiveness of the dynamic learning afforded by the wavelet coefficients, readers are invited to check our previous work [50] where we have presented a detailed discussion about it.



	Homo	Coarse	Fine	Nucl	Centro
Homo	94.96	1.63	2.86	0.53	0
Coarse	1.45	91.23	1.02	2.60	3.49
Fine	2.82	1.55	95.85	0.10	0
Nucl	0.41	2.18	0.24	94.60	1.59
Centro	0.36	3.41	0.03	2.17	94.92

(**a**) (**b**)

Figure 10. Results of the initial learning process: (**a**) the progression of the accuracy of training and validation; (**b**) the confusion matrix (total accuracy of 94.31%). "Homo", "Coarse", "Fine", Nucl" and "Centro" stand for homogeneous, coarse speckled, fine speckled, nucleolar and centromere, respectively.

After we had our pretrained model in hands, we could utilize it for the active learning over the LSHEp-2 dataset. For all the fine-tuning procedures, we had fixed all the layers before the second residual block, which means from the first convolutional layer to the second pooling layer (see Figure 5 for the architecture). The second residual block is set to be trainable because we want the networks to extract features that are specific to our main dataset before the feature fusion (layer concatenation). The final layer was changed to have six neurons, according the six classes of the LSHEp-2 dataset. The same learning approach was used for all the fine-tuning processes: a learning rate of 0.001 was utilized, training is stopped when the loss does not decrease five times in a row.

As said before, the LSHEp-2 dataset contains 63,445 images. A 80–20% splitting was performed, which gives 50,758 images for training, and 12,687 for testing. The labeling limitations that we imposed to ourselves only concern the training set (the testing set was just used for validation, not for fine-tuning). We principally tested our method with the limitation of being able to annotate only 20% of the training set, which gives a total of 10,152 images. The value *k* was set to be 1500, i.e., 1500 images (around 15% of the 10,152) were first selected randomly in order to perform the first fine-tuning. Additionally, then, the value m was set to be 1000, which means that we selected the first 1000 data in the ranking performed in step 4 (see Table 1) in every labeling iteration. We stop the labeling process after we annotated the totality of the 10,152 images. Note that the same kind of process was repeated for any level of limitation (the number k being 15% of the limitation and m being 10%).

The results are shown in two categories: the results without cross-modal transfer learning and the ones with cross-modal transfer learning. The first category designates the case where we did not use the initial learning for building the pretrained model. We just trained the networks by using directly the first k images from the main dataset. The second category designates the proposed scheme, where initial learning with a small dataset was used before fine-tuning with the main dataset. In every category, we show two cases for the results: using random sampling and using active learning-based sampling. In other words, and with the case of 20% of the limitation, random sampling designates the fact of selecting randomly 20% of the training images in order to train the networks while active learning-based sampling designates the fact of using active learning techniques for the selection of 20% of the training images.

Note that for all the cases where active learning was involved, we did not show the loss and accuracy progression since several different learning procedures were conducted in every labeling iteration (many fine-tunings). In this case, showing the loss and accuracy evolution was meaningless. On the other hand, these evolutions are shown for the cases that did not involve active learning, where only one single training procedure was performed. For simplicity, the different cases were designated by their short names shown in Table 2.

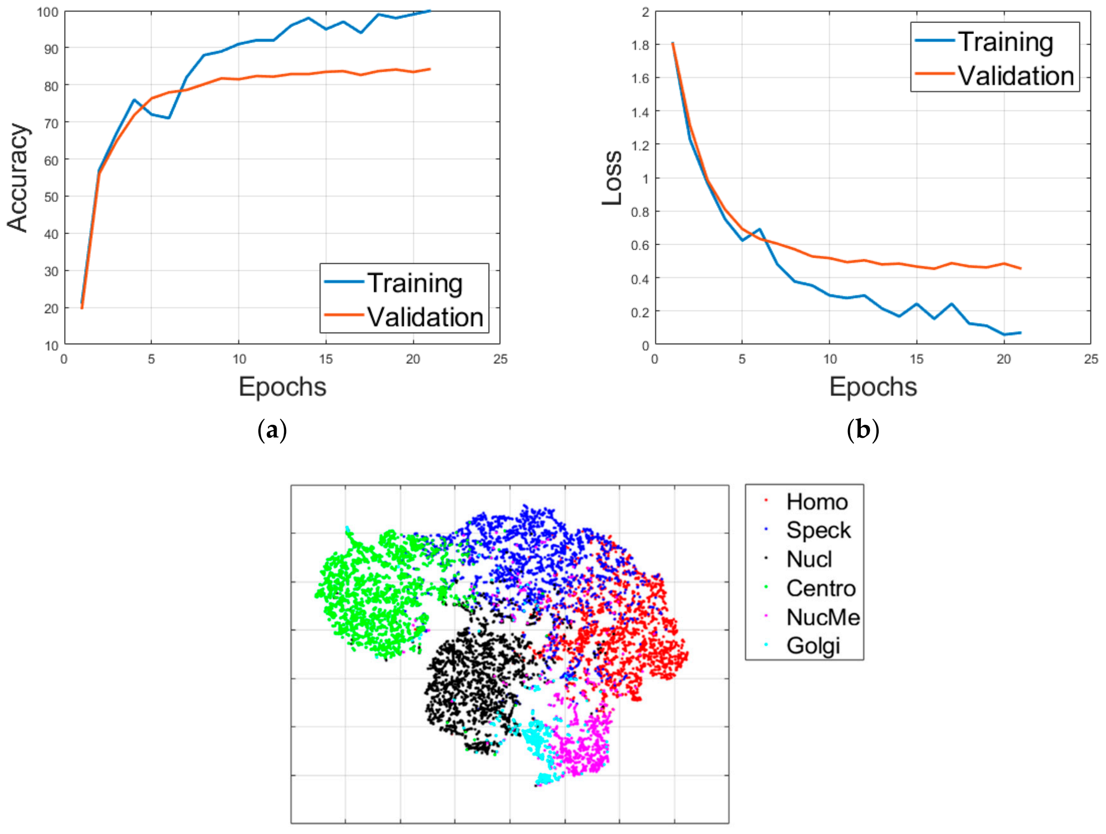
Table 2. The different cases used during the experiments.

Case Name	Comments
RS (random sampling)	No initial learning, and selection using *random sampling*.
AL (active learning)	No initial learning, and selection using *active learning*.
IN-RS (random sampling with cross-modal transfer learning)	Initial learning involved, and selection using *random sampling*.
IN-AL (active learning with cross-modal transfer learning)	Initial learning involved, and selection using *active learning*.

For other limitations (5%, 10%, 30%, 40% or even 100% of the training set), the results were summarized and discussed later. The case of 100% means that we could utilize the totality of the training data without any limitation. Note that the datasets (SNPHEp-2, 13A and LSHEp-2) all exist in a labeled form. The labeling limitations suggested in this work are indicative of the potential afforded by active learning and were used here in order to demonstrate its effectiveness.

3.2. Results without Cross-Modal Transfer Learning

As said before, all of the following results concern the case of 20% of the limitation. Figure 11 shows the detailed results of the "RS" case. This case just consists of selecting randomly 20% of the images and uses them to train the deep networks. Figure 11a shows the accuracy evolution over the training and validation sets (21 epochs). Figure 11b shows the loss evolution for the two sets. Figure 11c shows the visualization of the high-level features learned by the deep networks. All the visualizations here are obtained using the t-distributed stochastic neighbor embedding (t-SNE). Finally, Figure 11d shows the confusion matrix of the classification over the validation set. In all the confusion matrices shown here, "Homo", "Speck", "Nucl", "Centro", "NucMe" and "Golgi" refer to the homogeneous, speckled, nucleolar, centromere, nuclear membrane and Golgi cells, respectively.



(**a**) (**b**)

(**c**)

		Target Class					
		Homo	**Speck**	**Nucl**	**Centro**	**NucMe**	**Golgi**
Output Class	**Homo**	**89.69**	14.50	2.88	0	7.49	1.07
	Speck	8.13	**73.22**	1.51	1.41	8.23	2.93
	Nucl	0.91	4.43	**90.38**	0.55	3.44	9.60
	Centro	0.35	7.46	3.39	**98.00**	1.47	2.93
	NucMe	0.61	0.30	1.51	0	**46.68**	81.87
	Golgi	0.31	0.09	0.33	0.04	32.69	**1.60**

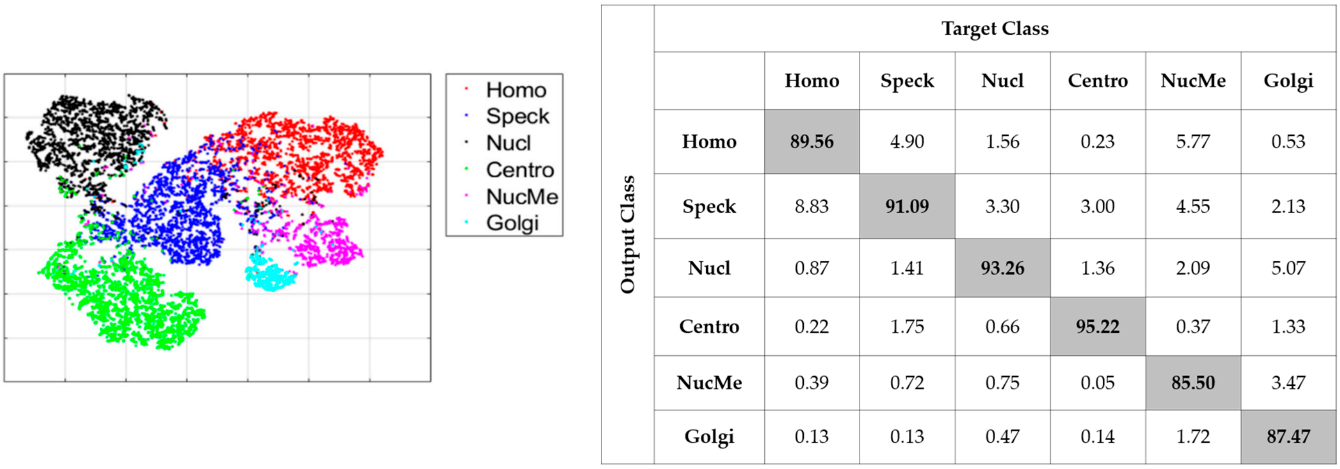
(**d**)

Figure 11. Classification results of the "RS" case: (**a**) accuracy, (**b**) loss, (**c**) features' visualization and (**d**) confusion matrix.

By analyzing the results, we could see that selecting randomly the data did not help for the generalization over the validation set. Two main observations can be highlighted from these results. First, there was a clear difference between the mean class accuracy (MCA) and the average classification accuracy (ACA). The MCA was 66.59% and the ACA was 81.13%. The ACA, which computes the overall accuracy by dividing the number of correctly classified data by the total number of data, appeared to take advantage of some of the classes that were very well discriminated. In particular, the nucleolar (90.38%) and the centromere (98.00%) contributed highly to establish the ACA in a very pleasant level.

On the other hand, the MCA, which computes the mean of all the classes' accuracies, was hugely impacted by the poor classification accuracy of the Golgi and nuclear membrane cells. As part of the second main observation, as we can also remark in Figure 11c by analyzing the visualization of the features, there was an extreme confusion between the two cells' clusters (Golgi in magenta and nuclear membrane in cyan). In fact, only 1.6% of the Golgi were well classified, while 81.87% of them were misclassified as the nuclear membrane, as we can see in the confusion matrix depicted in Figure 11d. Also, only 46.68% of the nuclear membrane were well classified. The two cellular types were certainly the most complicated to discriminate. The first reason is that both types are always underrepresented among the available data in all the existing datasets. There were only 375 Golgi and 814 nuclear membrane instances in the training set, while all the others cell types contained each at least 2100 images. The second reason is the complexity of their shape. Having the possibility of using only 20% of the training set, which diminishes again their number among the selected data for training, contributes to making the discrimination harder for the two cells. This pointed fact represents the principal observation of the present work. While having a limited number of labeled data in hands, the classification of these two cell types (Golgi and nuclear membrane) becomes really complex.

Figure 12 shows the results for the second case ("AL"). As for the first one, this case consists of not using the initial learning but, on the contrary, selects 20% of the data with active learning. The MCA for this case was 90.35% and the ACA was 91.51%. As we can notice in the confusion matrix in Figure 12b, most of the cells maintained a quite fair classification result. Furthermore, and even more importantly, the huge confusion between the Golgi (87.47% of accuracy) and nuclear membrane (85.50%) had clearly diminished. The visualization in Figure 12a shows a noticeable separation of the two clusters compared with Figure 11c. We could notice, in these results, the improvement afforded by the active learning-based selection. By selecting, precisely for annotation, the data for which the networks were the most confused about, active learning decreased the discrimination's complexity of the most difficult cells. At the same time, it maintained a good accuracy for the other cellular types.



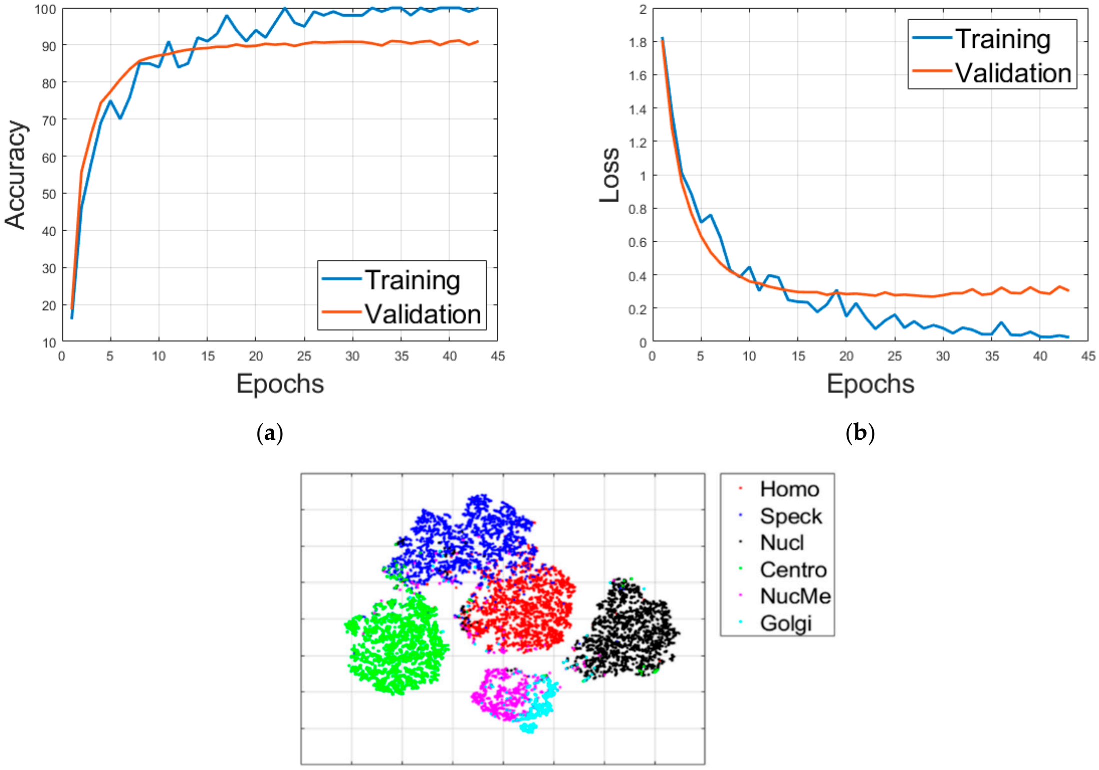
Output Class \ Target Class	Homo	Speck	Nucl	Centro	NucMe	Golgi
Homo	89.56	4.90	1.56	0.23	5.77	0.53
Speck	8.83	91.09	3.30	3.00	4.55	2.13
Nucl	0.87	1.41	93.26	1.36	2.09	5.07
Centro	0.22	1.75	0.66	95.22	0.37	1.33
NucMe	0.39	0.72	0.75	0.05	85.50	3.47
Golgi	0.13	0.13	0.47	0.14	1.72	87.47

(**a**) (**b**)

Figure 12. Classification results of the "AL" case: (**a**) features' visualization and (**b**) confusion matrix.

3.3. *Results with Cross-Modal Transfer Learning*

Here, we discuss the results obtained when an initial learning was performed in order to build the pretrained model. The first case ("IN-RS") consists of selecting the 20% randomly in order to perform fine-tuning. In Figure 13a–d, we show, respectively, the accuracy, loss, visualization of the features and confusion matrix.



(**a**)

(**b**)

(**c**)

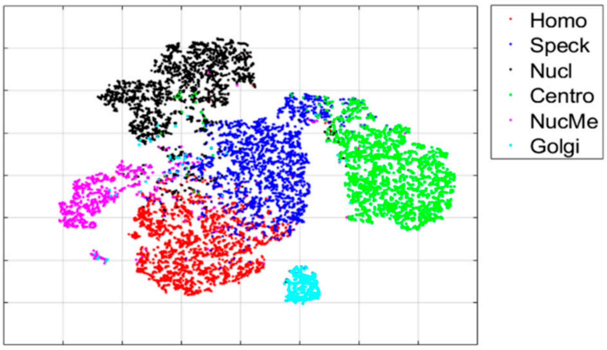
		Target Class					
		Homo	**Speck**	**Nucl**	**Centro**	**NucMe**	**Golgi**
Output Class	**Homo**	**91.52**	4.10	0.80	0.10	5.04	1.06
	Speck	6.83	**91.64**	1.37	1.09	4.18	1.60
	Nucl	1.13	1.24	**97.12**	0.77	3.32	8.27
	Centro	0.30	2.47	0.38	**98.04**	0.98	1.87
	NucMe	0.13	0.34	0.05	0	**50.86**	85.87
	Golgi	0.09	0.21	0.28	0	35.62	**1.33**

(**d**)

Figure 13. Classification results of the "IN_RS" case: (**a**) accuracy, (**b**) loss, (**c**) features' visualization and (**d**) confusion matrix.

Here also, we could notice how the extreme confusion remained present even after we applied cross-modal transfer learning. The MCA was 71.75%, which was better than the "RS" case. However, the poor accuracy (1.33%) of the Golgi really pulled down the MCA, even though the other cells accomplish excellent accuracies (50.86% for the nuclear membrane). The ACA was 87.54% for this case. Note that the initial learning process increased the overall accuracy (the ACA goes from 81.13% to 87.54% between "RS" and "IN-RS"), but not for the two most difficult cellular types.

The second case, denoted as "IN-AL", consists of selecting the 20% with active learning in order to perform fine-tuning. In Figure 14a,b, we have, respectively, the visualization of the features and the confusion matrix. As for the "AL" case discussed previously, we can notice how active learning permitted to tackle the extreme confusion between the Golgi and the nuclear membrane cells. This can be noticed in Figure 14a with the two clusters being completely disjoint, and in Figure 14b, where we see that the two cells accomplish reasonable accuracy. For this case, the MCA was 91.76% and the ACA was 92.77%. The most important observation here was that active learning strongly minimized the divergence between the two metrics by assuring a quite fair discrimination for all of the cellular types. Note that all of these results were obtained by using only 20% of the available training instances in the large HEp-2 dataset.



(**a**)

Output Class		Target Class					
		Homo	Speck	Nucl	Centro	NucMe	Golgi
	Homo	92.04	5.37	1.84	0.09	5.53	0.53
	Speck	6.61	90.50	1.89	1.68	2.58	2.13
	Nucl	0.87	1.36	94.01	1.14	1.72	5.07
	Centro	0.13	2.05	1.04	97.04	0.36	1.33
	NucMe	0.26	0.68	0.89	0	88.70	2.67
	Golgi	0.09	0.04	0.33	0.05	1.11	88.27

(**b**)

Figure 14. Classification results of the "IN-AL" case: (**a**) features' visualization and (**b**) confusion matrix.

In Figure 15, we show the classification accuracies of the three most difficult cells for the four cases ("RS", "AL", "IN-RS" and "IN-AL"). We can notice how both cases that use active learning, by allowing one to prioritize the annotation of the most difficult cells, "correct" the classification accuracies of the cases without active learning (especially for the Golgi and nuclear membrane).

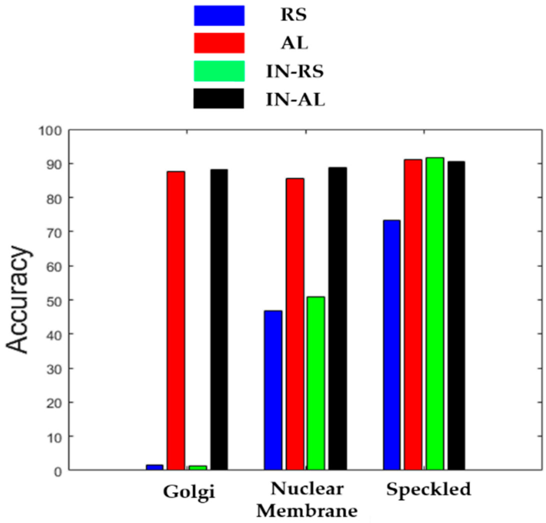


Figure 15. Classification accuracy of the three most difficult cells (the Golgi, the nuclear membrane and the speckled).

In Figure 16, we show the summary of the results (ACA) for the others limitations. We show for the 10%, 20% (discussed in details previously), 40%, 60%, 80% and 100%. As explained before, the 100% case refers to the fact of using all the available training data. There was no active learning process in this case since all the data were supposed to be labeled. In this case, and only for this case, random sampling and active learning results were the same, as we can notice in Figure 16. For all the other limitations, we can see how active learning can help to achieve satisfying results even though we do not have access to the totality of the training data. In fact, for all the limitations, active learning-based labeling provided accuracies that were superior to 90%. It is only for the case of 10% limitation that active learning without cross-modal transfer learning ("AL") achieved 86.68% (see Figure 16). However, this result could be significantly improved by using cross-modal transfer learning, as proposed in this work. In that case ("IN-AL"), the accuracy for the 10% limitation reached 90.23%. In other words, active learning coupled with cross-modal transfer learning allows one to achieve satisfying discrimination results even with a few number of labeled data in hands.

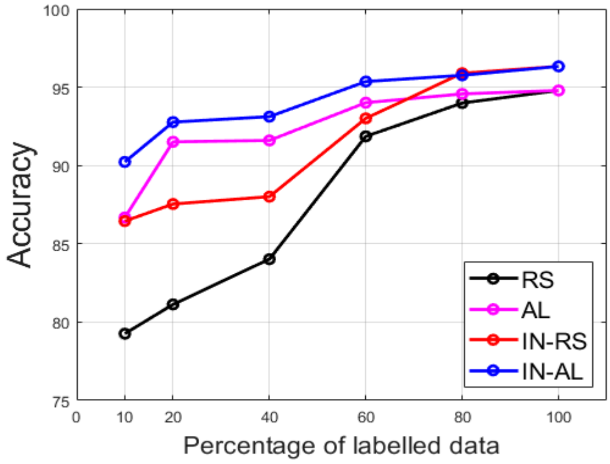


Figure 16. Evolution of the accuracy with other limitations in terms of labeling. The accuracy is shown for the 4 different cases discussed above.

In order to show the contribution of cross-modal transfer learning, we show in parallel the classification accuracy when there was no limitation (100% of training data available) for the case where no initial learning was performed and for the case where we used the small dataset in order to build the pretrained model and perform cross-modal transfer learning. Figure 17 shows the comparison. We can remark how using the initial learning really improved the overall accuracy of the networks.

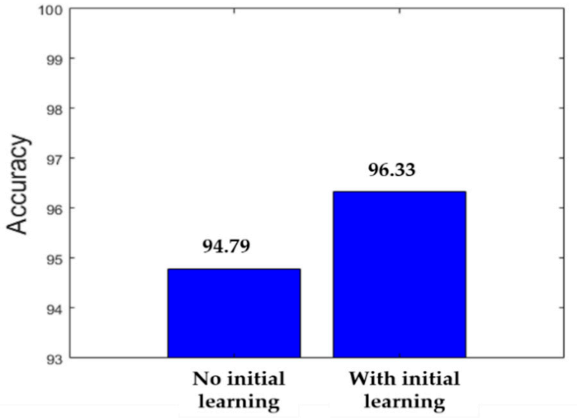


Figure 17. Accuracy improvement with the initial learning in the case of 100% of the training data available.

In Table 3, we show the results of some different approaches. Note that most of the approaches were proposed for either the ICPR2012 or 13A datasets. In this comparative study, we tried to see how these approaches reacted on the bigger and more complex LSHEp-2 dataset. Note also that none of these approaches utilized active learning. The comparisons were done with the use of the totality of the training data and the aim of the comparative study was more to demonstrate the effectiveness of our deep parallel networks and the contribution of the cross-modal transfer learning process.

Table 3. Comparative study using the LSHEp-2 dataset.

Methods	Accuracy (ACA)
Handcrafted features-based approach [60]	86.61%
LeNet-5-like CNN without transfer learning [16]	88.75%
VGG-16-like network without transfer learning [21]	90.23%
Transfer learning using the pretrained VGG-19	91.57%
Transfer learning using the pretrained AlexNet	92.41%
Transfer learning using the pretrained VGG-16 [32]	92.89%
DCR-Net [25]	94.15%
Transfer learning using the pretrained ResNet-50	94.36%
Our proposed deep parallel residual nets without cross-modal transfer learning	94.79%
Cross-modal transfer learning using ResNet-50 [38]	95.94%
Our proposed deep parallel residual nets with cross-modal transfer learning	96.33%

Similar hyperparameters' settings were used for the training procedures of the models used here for comparison. The learning rate was set to 0.001 and training was terminated after the loss plateaus for 10 consecutive epochs. We used a mini-batch of 128. Except in the case of our method, for all the methods involving transfer learning, the parameters were updated for all the layers, since all of the pretrained models were previously trained on ImageNet. Data augmentation, using the same technique as previously explained, was applied for the two methods that did not involve transfer learning, since the models were trained from the scratch.

As we can see in Table 3, most of the approaches using the models that were pretrained on ImageNet performed less than the ones that used cross-modal transfer learning. The state-of-the-art method in [38] utilize ResNet-50 but with an initial learning performed by using the ICPR2012 dataset, while their targeted dataset is the 13A. Our proposed method uses the deep parallel networks and the SNPHEp-2 dataset was utilized for the initial learning. Another state-of-the-art method is the DRC-Net [25], which achieves 94.15% on the LSHEP-2 dataset. We can notice that the accuracies shown in Figure 16 are similar with these state-of-the-art performances. Active learning coupled with cross-modal transfer learning allows one to achieve pleasant performance even with limitations in terms of labeling.

Note that models like VGG-16, VGG-19 and AlexNet require a substantial memory because of the enormous number of parameters generated by the fully connected layers at the end of the network. The residual networks (our parallel networks and the ResNet-50) also require a lot of memory but the computational complexity is far less compared to the other networks. This is explained by the global averaging pooling layer, which efficiently minimizes the computational complexity by diminishing the total number of parameters of the networks. Another important point to note is that all the methods used in the comparative study in Table 3 necessitate only one single training procedure. On the other hand, every case that involves active learning in our method necessitates several training procedures because of the iterative labeling process. This fact can be considered as the principal limitation of our method, as the cascade of training procedures elongate the time needed to build the final model. However, as previously discussed, our aim was to demonstrate that we could achieve quite pleasant performance with only a limited number of labeled data in hands.

4. Conclusions

The automatic classification of the HEp-2 cell images represents an essential step in the production of the computer-aided diagnosis systems. The quasi-totality of the approaches in the HEp-2 cell classification literature prefer to address this problem by adopting the supervised learning approaches. Deep learning-based supervised learning necessitates the availability of thousands of images labeled by hands by the biological experts. This labeling process, especially in the case of big datasets, can represent a time consuming and burdensome task. Our work aimed to present a supervised learning approach that can minimize the need of the labeled data, thus, minimize the cost related to the labeling process.

In this purpose, we proposed a methodology that utilizes active learning coupled with transfer learning. We first proposed deep parallel networks that tackle the inter-class variations of the HEp-2 datasets. We then proposed an initial learning process using these networks over a quite small and labeled dataset in order to have a pretrained model. This pretrained model was then used with active learning on the targeted dataset, which was much bigger and more complex than most of the popular HEp-2 datasets, in order to select only the data that really need to be labeled.

The proposed methodology alleviated the labeling process by allowing the networks to achieve quite satisfying discrimination results even with a limited number of labeled data. In fact, by using only 20% of the available training examples, our parallel networks achieved a mean class accuracy (MCA) of 91.76% while the average classification accuracy (ACA) reached 92.77%. These results were similar with the performance reached by the actual state-of-the-art methods, which, on the other hand, utilize the totally of the available training data. We demonstrated that this was made possible by the fact that active learning allows to prioritize the labeling of the most difficult cells and, thus, allowed the networks to maintain a good discrimination performance for each one of the cellular types. We demonstrated that the proposed methodology allowed us to minimize the divergence between the MCA and ACA by assuring a better discrimination for the Golgi and nuclear membrane cells. Since the need of data increased exponentially with the deep learning-based applications, we believed that the present work could be useful in the future in order to somehow ease the manual labeling task. The next step of our work consists of exploring different selection scenarios that do not necessitate training the model several times in order to reduce the computational complexity.

Author Contributions: Conceptualization, C.V.; Funding acquisition, S.-H.L. and K.-R.K.; Investigation, C.V.; Methodology, C.V.; Project administration, S.-H.L. and K.-R.K.; Software, C.V., S.-H.L. and K.-R.K.; Supervision, S.-H.L. and K.-R.K.; Validation, S.-H.L. and K.-R.K.; Writing—original draft, C.V.; Writing—review and editing, C.V and S.-H.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Brain Korea 21 project (BK21).

Institutional Review Board Statement: Not Applicable.

Informed Consent Statement: Not Applicable.

Data Availability Statement: All the data used for the experiments are publicly available can at http://qixianbiao.github.io/HEp2Cell/ (accessed on 16 January 2021).

Acknowledgments: This research was supported by the Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education (2020R1I1A306659411, 2020R1F1A1069124), and Ministry of Trade, Industry and Energy for its financial support of the project titled "the establishment of advanced marine industry open laboratory and development of realistic convergence content".

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Rigon, A.; Soda, P.; Zennaro, D.; Iannello, G.; Afeltra, A. Indirect immunofluorescence in autoimmune diseases: Assessment of digital images for diagnostic purpose. *Cytom. B Clin. Cytom.* **2007**, *72*, 472–477. [CrossRef] [PubMed]
2. Foggia, P.; Percannella, G.; Soda, P.; Vento, M. Benchmarking HEp-2 cells classification methods. *IEEE Trans. Med. Imaging* **2013**, *32*, 1878–1889. [CrossRef] [PubMed]
3. Foggia, P.; Percannella, G.; Saggese, A.; Vento, M. Pattern recognition in stained HEp-2 cells: Where are we now? *Pattern Recognit.* **2014**, *47*, 2305–2314. [CrossRef]
4. Cataldo, S.D.; Bottino, A.; Ficarra, E.; Macii, E. Applying textural features to the classification of HEp-2 cell patterns in IIF images. In Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012), Tsukuba, Japan, 11–15 November 2012; pp. 689–694.
5. Wiliem, A.; Wong, Y.; Sanderson, C.; Hobson, P.; Chen, S.; Lovell, B.C. Classification of human epithelial type 2 cell indirect immunofluorescence images via codebook based descriptors. In Proceedings of the 2013 IEEE Workshop on Applications of Computer Vision (WACV), Tampa, FL, USA, 15–17 January 2013; pp. 95–102. [CrossRef]
6. Stoklasa, R.; Majtner, T.; Svoboda, D. Efficient k-NN based HEp-2 cells classifier. *Pattern Recognit.* **2014**, *47*, 249–2418. [CrossRef]
7. Nosaka, R.; Fukui, K. HEp-2 cell classification using rotation invariant co-occurrence among local binary patterns. *Pattern Recognit.* **2014**, *47*, 2428–2436. [CrossRef]
8. Cataldo, S.D.; Bottino, A.; Islam, I.U.; Vieira, T.F.; Ficarra, E. Subclass discriminant analysis of morphological and textural features for HEp-2 staining pattern classification. *Pattern Recognit.* **2014**, *47*, 2389–2399. [CrossRef]
9. Theodorakopoulos, I.; Kastaniotis, D.; Economou, G.; Fotopoulos, S. HEp-2cells classification via sparse representation of textural features fused into dissimilarity space. *Pattern Recognit.* **2014**, *47*, 2367–2378. [CrossRef]
10. Huang, Y.C.; Hsieh, T.Y.; Chang, C.Y.; Cheng, W.T.; Lin, Y.C.; Huang, Y.L. HEp-2 cell images classification based on textural and statistic features using self-organizing map. In Proceedings of the 4th Asian Conference on Intelligent Information and Database Systems, Part II, Kaohsiung, Taiwan, 19–21 March 2012; pp. 529–538.
11. Thibault, G.; Angulo, J.; Meyer, F. Advanced statistical matrices for texture characterization: Application to cell classification. *IEEE Trans. Biomed. Eng.* **2014**, *61*, 630–637. [CrossRef] [PubMed]
12. Wiliem, A.; Sanderson, C.; Wong, Y.; Hobson, P.; Minchin, R.F.; Lovell, B.C. Automatic classification of human epithelial type 2 cell indirect immunofluorescence images using cell pyramid matching. *Pattern Recognit.* **2014**, *47*, 2315–2324. [CrossRef]
13. Xu, X.; Lin, F.; Ng, C.; Leong, K.P. Automated classification for HEp-2 cells based on linear local distance coding framework. *J. Image Video Proc.* **2015**, *2015*, 1–13. [CrossRef]
14. Ponomarev, G.V.; Arlazarov, V.L.; Gelfand, M.S.; Kazanov, M.D. ANA HEp-2 cells image classification using number, size, shape and localization of targeted cell regions. *Pattern Recognit.* **2014**, *47*, 2360–2366. [CrossRef]
15. Shen, L.; Lin, J.; Wu, S.; Yu, S. HEp-2 image classification using intensity order pooling based features and bag of words. *Pattern Recognit.* **2014**, *47*, 2419–2427. [CrossRef]
16. Gao, Z.; Wang, L.; Zhou, L.; Zhang, J. HEp-2 cell image classification with deep convolutional neural networks. *IEEE J. Biomed. Health Inf.* **2017**, *21*, 416–428. [CrossRef]
17. Bayramoglu, N.; Kannala, J.; Heikkilä, J. Human epithelial type 2 cell classification with convolutional neural networks. In Proceedings of the IEEE 15th Intrenational Conference on Bioinformatics and Bioengineering (BIBE), Belgrade, Serbia, 2–4 November 2015; pp. 1–6.
18. Jia, Y.; Shelhamer, E.; Donahue, J.; Karayev, S.; Long, J.; Girshick, R.; Guadarrama, S.; Darrell, T. Caffe: Convolutional architecture for fast feature embedding. In Proceedings of the 22nd ACM International Conference on Multimedia, Orlando, FL, USA, 3–7 November 2014; pp. 675–678. [CrossRef]
19. Liu, J.; Xu, B.; Shen, L.; Garibaldi, J.; Qiu, G. HEp-2 cell classification based on a deep autoencoding-classification convolutional neural network. In Proceedings of the 14th IEEE International Symposium on Biomedical Imaging (ISBI 2017), Melbourne, Australia, 18–21 April 2017; pp. 1019–1023. [CrossRef]
20. Rodrigues, L.F.; Naldi, M.C.; Mari, J.F. Comparing convolutional neural networks and preprocessing techniques for HEp-2 cell classification in immunofluorescence images. *Comput. Biol. Med.* **2020**, *116*, 103542. [CrossRef]
21. Xi, J.; Linlin, S.; Xiande, Z.; Shiqi, Y. Deep convolutional neural network based HEp-2 cell classification. In Proceedings of the 2016 23rd International Conference on Pattern Recognition (ICPR), Cancun, Mexico, 4–8 December 2016; pp. 77–80. [CrossRef]
22. Li, Y.; Shen, L. A deep residual inception network for HEp-2 cell classification. In Proceedings of the3rd International Workshop, DLMIA 2017, and 7th International Workshop, ML-CDS 2017, Québec City, QC, Canada, 14 September 2017.
23. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
24. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P. Going deeper with convolutions. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 1–9.
25. Shen, L.; Jia, X.; Li, Y. Deep cross residual network for HEp-2 cell staining pattern classification. *Pattern Recognit.* **2018**, *82*, 68–78. [CrossRef]

26. Majtner, T.; Bajic, B.; Lindblad, J.; Sladoje, N.; Blanes-Vidal, V.; Nadimi, E.S. On the effectiveness of generative adversarial networks as HEp-2 image augmentation tool. In Proceedings of the Scandinavian Conference on Image Analysis (SCIA 2019), Norrkoping, Sweden, 11–13 June 2019; pp. 439–451. Available online: https://link.springer.com/chapter/10.1007%2F978-3-030-20205-7_36 (accessed on 16 January 2021).
27. Goodfellow, I.J.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative adversarial nets. In Proceedings of the 27th International Conference on Neural Information Processing Systems (NIPS 2014), Montréal, QC, Canada, 8–13 December 2014; pp. 2672–2680. Available online: https://arxiv.org/pdf/1406.2661.pdf (accessed on 16 January 2021).
28. Radford, A.; Metz, L.; Chintala, S. Unsupervised representation learning with deep convolutional generative adversarial networks. In Proceedings of the 4th International Conference on Learning Representations (ICLR 2016), San Juan, Puerto Rico, 2–4 May 2016; Available online: https://arxiv.org/pdf/1511.06434.pdf (accessed on 16 January 2021).
29. Li, Y.; Shen, L. HEp-Net: A smaller and better deep-learning network for HEp-2 cell classification. *Comput. Methods Biomech. Biomed. Eng.* **2019**, *7*, 266–272. [CrossRef]
30. Vununu, C.; Lee, S.-K.; Kwon, K.-R. A Deep feature extraction method for HEp-2 image classification. *Electronics* **2018**, *8*, 20. [CrossRef]
31. Vununu, C.; Lee, S.-K.; Kwon, K.-R. A strictly unsupervised deep learning method for HEp-2 cell image classification. *Sensors* **2020**, *20*, 2717. [CrossRef]
32. Phan, H.T.H.; Kumar, A.; Kim, J.; Feng, D. Transfer learning of a convolutional neural network for HEp-2 cell image classification. In Proceedings of the 2016 IEEE 13th International Symposium on Biomedical Imaging (ISBI), Prague, Czech Republic, 16 June 2016; pp. 1208–1211.
33. Simonyan, K.; Zisserman, A. A very deep convolutional networks for large-scale image recognition. In Proceedings of the 2015 International Conference on Learning Representation (ICLR15), San Diego, CA, USA, 7–9 May 2015; Available online: https://arxiv.org/pdf/1409.1556.pdf (accessed on 16 January 2021).
34. Lu, M.; Gao, L.; Guo, X.; Liu, Q.; Yin, J. HEp-2 cell image classification method based on very deep convolutional networks with small datasets. In Proceedings of the 9th International Conference on Digital Image Processing (ICDIP 2017), Hong Kong, China, 19–22 May 2017. [CrossRef]
35. Nguyen, L.D.; Gao, R.; Lin, D.; Lin, Z. Biomedical image classification based on a feature concatenation and ensemble of deep CNNs. *J. Ambient Intell. Hum. Comput.* **2019**. [CrossRef]
36. Cascio, D.; Taormina, V.; Raso, G. Deep CNN for IIF images classification in autoimmune diagnostics. *Appl. Sci.* **2019**, *9*, 1618. [CrossRef]
37. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet classification with deep convolutional neural networks. In Proceedings of the 25th International Conference on Neural Information Processing Systems (NIPS'12), Lake Tahoe, NV, USA, 3–6 December 2012; pp. 1097–1105.
38. Lei, H.; Han, T.; Zhou, F.; Yu, Z.; Qin, J.; Elazab, A.; Lei, B. A deeply supervised residual network for HEp-2 cell classification via cross-modal transfer learning. *Pattern Recognit.* **2018**, *79*, 290–302. [CrossRef]
39. Lovell, B.C.; Percannella, G.; Saggese, A.; Vento, M.; Wiliem, A. International contest on pattern recognition techniques for indirect immunofluorescence images analysis. In Proceedings of the 2016 23rd International Conference on Pattern Recognition (ICPR), Cancun, Mexico, 4–8 December 2016; pp. 74–76.
40. Lewis, D.D.; Catlett, J. Heterogeneous uncertainty sampling for supervised learning. In Proceedings of the Eleventh International Conference on Machine Learning, New Brunswick, NJ, USA, 10–13 July 1994; pp. 148–156. [CrossRef]
41. Seung, H.S.; Opper, M.; Sompolinsky, H. Query by committee. In Proceedings of the Fifth Annual Workshop on Computational Learning Theory, Pittsburgh, PA, USA, 27–29 July 1992; pp. 287–294. [CrossRef]
42. Shi, F.; Wang, Z.; Hu, M.; Zhai, G. Active learning plus deep learning can establish cost-effective and robust model for multichannel image: A case of hyperspectral image classification. *Sensors* **2020**, *20*, 4975. [CrossRef]
43. Wang, D.; Shang, Y. A new active labeling method for deep learning. In Proceedings of the 2014 International Joint Conference on Neural Networks (IJCNN), Beijing, China, 6–11 July 2014; pp. 112–119. [CrossRef]
44. Zhou, Z.; Shin, J.; Zhang, L.; Gurudu, S.; Gotway, M.; Liang, J. Fine-tuning convolutional neural networks for biomedical image analysis: Actively and incrementally. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 7340–7351. [CrossRef]
45. Sener, O.; Savarese, S. Active learning for convolutional neural networks: A core-set approach. In Proceedings of the International Conference on Learning Representations, Vancouver, BC, Canada, 30 April–3 May 2018; Available online: https://arxiv.org/pdf/1708.00489.pdf (accessed on 16 January 2021).
46. Tkachenko, R.; Izonin, I. Model and principals for the implementation of neural-like structures based on geometric data transformations. *Adv. Intell. Syst. Comput.* **2019**, *754*, 578–587. [CrossRef]
47. Cascio, D.; Taormina, V.; Raso, G. Deep convolutional neural network for HEp-2 fluorescence intensity classification. *Appl. Sci.* **2019**, *9*, 408. [CrossRef]
48. Merone, M.; Sansone, C.; Soda, P. A computer-aided diagnosis system for HEp-2 fluorescence intensity classification. *Artif. Intell. Med.* **2019**, *97*, 71–78. [CrossRef]
49. Nigam, I.; Agrawal, S.; Singh, R.; Vatsa, M. Revisiting HEp-2 cell classification. *IEEE Access* **2015**, *3*, 3102–3113. [CrossRef]

50. Vununu, C.; Lee, S.-K.; Kwon, O.-J.; Kwon, K.-R. A dynamic learning method for the classification of the HEp-2 cell images. *Electronics* **2019**, *8*, 850. [CrossRef]
51. Ioffe, S.; Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In Proceedings of the 32nd International Conference on Machine Learning, Lille, France, 7–9 July 2015; pp. 448–456.
52. Bengio, Y. Learning deep architecture for AI. *Foundat. Trends Mach. Learn.* **2009**, *2*, 1–127. [CrossRef]
53. Rumelhart, D.E.; Hinton, G.E.; Williams, R.J. Learning representations by back-propagating errors. *Nature* **1986**, *323*, 533–536. [CrossRef]
54. Shannon, C.E. A mathematical theory of communication. *ACM Sigmob. Mob. Comput. Commun. Rev.* **2001**, *5*, 3–55. [CrossRef]
55. Lewis, D.D.; Gale, W.A. A sequential algorithm for training text classifiers. In Proceedings of the 17th ACM International Conference on Research and Development in Information Retrieval, Dublin, Ireland, 3–6 July 1994; pp. 3–12. [CrossRef]
56. Scheffer, T.; Decomain, C.; Wrobel, S. Active hidden markov models for information extraction. In *International Symposium on Intelligent Data Analysis*; Springer: Berlin/Heidelberg, Germany, 2001; pp. 309–318. [CrossRef]
57. Tong, S.; Koller, D. Support vector machine active learning with applications to text classification. *J. Mach. Learn. Res.* **2001**, *2*, 45–66. [CrossRef]
58. Abe, N.; Mamitsuka, H. Query learning strategies using boosting and bagging. In Proceedings of the Fifteenth International Conference on Machine Learning (ICML 1998), Madison, WI, USA, 24–27 July 1998; pp. 1–8.
59. Huo, L.Z.; Tang, P. A batch-mode active learning algorithm using region-partitioning diversity for SVM classifier. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2014**, *7*, 1036–1046. [CrossRef]
60. Qi, X.; Zhao, G.; Chen, J.; Pietikainen, M. Exploring illumination robust descriptors for human epithelial type 2 cell classification. *Pattern Recognit.* **2016**, *60*, 420–429. [CrossRef]