Towards Document Entity Recognition using Close Domain Transfer Learning

Adrián Inés, César Domínguez, Jónathan Heras, Eloy Mata and Vico Pascual
Department of Mathematics and Computer Science
Universidad de La Rioja
Logroño, Spain
{adrian.ines, cesar.dominguez, jonathan.heras, eloy.mata, mvico}@unirioja.es

***Abstract*—Nowadays, document data analysis is an increasing important task due to the large number of generated documents. Due to the increasing volume of documents, their manual analysis is a tedious and time-consuming process; and, therefore, automatic methods have arisen to deal with this task. Nowadays, a feasible approach for automatic document analysis is the application of deep learning methods. However, using these techniques requires a large number of annotated images, which can be difficult to obtain. This problem can be solved by using transfer learning, a technique that allows us to reuse the knowledge acquired from a different task. In this work, we have studied three different transfer learning approaches to create object detection models for the recognition of document entities such as text, titles, tables or figures. First, we have applied transfer learning from models pretrained in general detection tasks related to natural images. Second, we have studied an approach by training the backbones of the detection models in a close domain, using a document classification dataset. Finally, we have performed a close domain transfer learning training; that is, we have trained a model in a document detection dataset, and, then, we have retrained it in the target task. We have carried out a thorough analysis of each of the 3 approaches using 3 different detection architectures, and 5 document detection datasets. Our results show that, the close domain transfer learning approach can improve the performance of models between a 3% and a 20%.**

***Index Terms*—Object Detection, Transfer learning, Deep Learning, Document images.**

I. Introduction

Document analysis has become increasingly important due to the rapid growth in the number of available documents [4]. Every day thousands of documents, such as reports, forms, emails or invoices are either manually or automatically generated. Among the document analysis tasks, we can distinguish the classification of documents [3]; the detection of document entities, such as text, titles, tables or figures, within documents [11]; or the segmentation of document entities [5]. When the volume of documents increases, the manual analysis of documents becomes a very demanding task both in time and human resources. For that reason, analysing documents automatically is instrumental to exploit the information that is stored in them [6].

This work was partially supported by Ministerio de Economía y Competitividad [MTM2017-88804-P], and ADER [2017-I-IDD-00018], and Ministerio de Ciencia e Innovación and FEDER Funds [RTC-2017-6640-7], and FPU Grant 16/06903 of the Spanish MEC.

Currently, we can find different automatic techniques for document analysis. Among these techniques, we can highlight Optical Character Recognition (OCR) [22], computer vision techniques [12], and natural language processing methods [35]. In addition, deep learning has been successfully employed for document analysis [28]. The main problem with deep learning techniques is the large amount of data that is required to make them work properly [1]. At first sight, this might not seem a problem, due to the large number of documents that are generated in a daily basis; however, the manual annotation of these documents, a step that is instrumental to use supervised deep learning techniques, requires a considerable amount of time.

In the context of document entity recognition, the main approach employed to deal with the lack of enough annotated images consists in reusing a previously pre-trained model [16], [18], [19], [34]. Such models are either trained using an unsupervised regime from unlabelled images and taking into account visual, textual and layout features [16], [34]; or by employing models trained on a different domain that are adjusted to a particular task using either visual, textual or layout features [18], [19]. The advantage of the former models is that they are specifically designed for document understanding tasks; however, they require lots of resources to be trained (for instance, LayoutLM takes 170 hours per epoch using 8 NVIDIA Tesla V100 32GB GPUs) and are generally difficult to employ. On the contrary, the latter models are based on widely employed and standard deep learning architectures that are not as computationally demanding as the former models, but whose performance is worse. In this paper, we investigate how we can improve the latter models by selecting an adequate initial domain.

Models trained on a different domain that are adjusted to perform a different task belong to the category of transfer learning methods [25]. This technique makes possible to reduce the number of annotated images needed to train a deep learning model to solve a particular task by re-using the knowledge acquired in a different task. The main drawback of this technique appears when the tasks come from far domains. In most cases, transfer learning is applied from models that have been pretrained in large natural image datasets, like, ImageNet [7] for classification, and COCO [20] and PASCAL-VOC [8] for object detection; but, transfer learning from

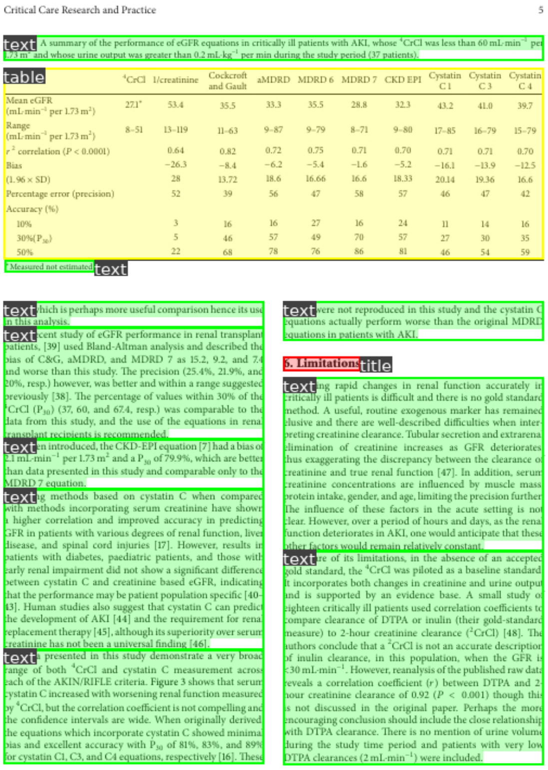

Fig. 1. Entity recognition on documents task example.

natural images to, for instance, medical images has several limitations due to the differences between these kinds of images [21], [24]. The hypothesis that we investigate in this work is that an adequate selection of a relevant source task can boost the performance of entity recognition on documents, see Figure 1.

The main contribution of this work is the study of different transfer learning approaches to create detection models for document entity recognition based on visual features. In particular, we have studied three different approaches that are presented in Section II. In Section III, we have explained the results obtained from the 3 approaches applied to 4 document entity recognition tasks. Finally, we end the paper drawing some conclusions from this study, and proposing new possible approaches in Section IV.

II. Materials and Methods

Object detection is a computer vision task that aims to locate the position of multiple objects in an image, and also provides the class of such objects. Currently, object detection tasks are mainly tackled using deep learning architectures that consist of a classification backbone and a head in charge of generating the detection boxes. Since training from-scratch these architectures is time-consuming and requires a considerable amount of resources, transfer learning is usually applied.

Transfer learning for object detection can be performed either by loading a pretrained backbone, or by loading the whole model. In both cases, the model is later fine-tuned after the loading step. In this work, we have studied 3 different transfer learning approaches (traditional transfer learning, close backbone transfer, and, close architecture transfer) for document entity recognition compared with a from-scratch training process. For our experiments, we have used 3 deep learning architectures that are FasterRCNN [27], EfficientDet [32] and YOLO v4 [2]. FasterRCNN and EfficientDet are implemented in PyTorch [23] and have been trained thanks to the functionality of the Fastai [17] and IceVision[1] libraries using a GPU Nvidia RTX 2080 Ti. YOLO v4 is implemented in Darknet [26] and has been trained using the Darknet utilities and the GPU specified before.

These architectures have been tested on 5 different document entity detection datasets. In particular, we have used a large dataset and 4 small datasets to test the performance of the various transfer learning approaches. In Table I, we can see a description of the 5 datasets and the number of images that we have used for training and testing. We briefly describe these datasets as follows. The PubLayNet dataset, is a large dataset for document layout analysis that contains images of research papers and articles and annotations for various elements in a page (text, title, list, tables and figures). This dataset contains $335,703$ training images and $11,405$ test images of size 512×512. The ICDAR2017 dataset [10] consists of $2,000$ images of scientific papers with 3 kinds of objects to be detected: formulas, figures and tables. The FUNSD dataset [13] consists of 199 fully annotated forms with 4 kind of objects to be detected: headers, questions, answers and others. The UNLV dataset [29] contains 427 examples in scanned image format in which tables have to been detected. And, the Marmot dataset [9], that contains $2,000$ pages in PDF format where tables from research papers have to be detected.

Using these datasets, we have trained the aforementioned models by using three different transfer learning approaches and a scratch approach, that is, with random initial weights for both, the detection head and the backbone of the detection architectures, see Figure 2. The rest of this section is devoted to present the three different transfer learning approaches.

A. Traditional transfer learning

In the first transfer learning approach, we have studied the classical transfer learning approach using models pretrained on natural images. Currently, we can find a lot of pretrained detection models, most of them are pretrained in general detection tasks, using datasets such as COCO [20] and Pascal-VOC [8]. Also, we can find detection models with a backbone pretrained in a general classification task using the ImageNet dataset [7]. These datasets have hundreds of thousands of natural images with numerous examples in each image. Therefore, our first approach consists in reusing the knowledge acquired by detection models in these large datasets.

[1] https://airctic.com/

Dataset	♯ Train	♯ Test	Classes	Entities	Format	Colour
PubLayNet [36]	$335,703$	$11,405$	5	Texts, Titles, Lists, Tables and Figures	JPG	Yes
ICDAR [10]	$1,200$	400	3	Formulas, Figures and Tables	JPG	Yes
FUNSD [13]	149	50	4	Headers, Questions, Answers and Others	PNG	No
UNLV [29]	302	101	1	Tables	JPG	No
Marmot [9]	754	252	1	Tables	JPG	Yes

TABLE I
DESCRIPTION OF THE DATASETS EMPLOYED IN OUR EXPERIMENTS AND THE SPLIT MADE FOR TRAINING AND TESTING.

In particular, we have considered detection models with a backbone pretrained in the ImageNet challenge and a detection head with randomly initialised weights, see Figure 2. Then, we have fine-tuned these models using the new dataset. The Faster-RCNN and EfficientDet models have been trained using a two-stage transfer-learning method similar to the one presented in [17]. Specifically, in a first stage, we freezed the pretrained backbone of the model and trained the head of the detection model for two epochs. In the second stage, we unfreezed the whole model and retrained the model with the new data for 15 epochs using a suitable learning rate. In particular, we selected the learning rate that decreases the loss to the minimum possible value using the algorithm presented in [31]. Moreover, we employed early stopping based on monitoring the valid loss, and data augmentation [30] (using flips, rotations, zooms and lighting transformations) to prevent overfitting. For the YOLO v4 model, we have trained it by loading the pretrained ImageNet backbone, and then fine-tuned the whole model by one step training process with $12,000$ steps. We used 1e-3 as learning rate, and also applied early stopping and data augmentation to avoid overfitting.

B. Close backbone transfer

As we have previously explained, detection models consist of a classification backbone, and a head that allows us to generate the detection boxes. The idea of our second transfer learning approach is to perform a two-stage training process. First, we have trained a classification model in a document classification task. And, in a second stage, we have employed the methods presented in the previous section, but using the specific backbone trained on the previous stage, see Figure 2.

For the first stage of the process, we have trained three different families of classification architectures that are used as backbones by the detection models: ResNet50 [15] (for the Faster-RCNN architecture), EfficientNet-b2 [33] (for the EfficientDet architecture) and CSPDarknet53 [2] (for the YOLO architecture). These models have been trained using a two-stage transfer learning procedure similar to the procedure explained in the previous section. In the first stage, we replaced the last layers of the model (that is, the layers that give us the classification of the images), with a new one adapted to the number of classes of each particular dataset. Then, we trained these new layers (the rest of the layers stayed frozen) for two epochs. In the second stage, we unfreezed the whole model and retrained all the layers of the model with the new data for 50 epochs using a suitable learning rate [31].

The dataset selected to train these models was the RVL-CDIP dataset [14], a document classification dataset with 16 classes that consists of $400,000$ grayscale images of size 256×331 with $25,000$ images per class. These images are split into $320,000$ training images, $40,000$ validation images, and $40,000$ test images. In our experiments we have used the $320,000$ training images to train the models and the $40,000$ validation images as testing set. Also, we have resized the images to size 512×512. The trained models obtained the following accuracy: ResNet50, 0.9116; EfficientNet-B2, 0.9227; and CSPDarknet53, 0.8946.

C. Close architecture transfer

The last approach consists in training a detection model pretrained on a document entity detection task. To this aim, we have trained a detection model in a document entity detection task where enough annotated images where available; and, subsequently, we have retrained such model with a smaller target dataset, see Figure 1. The goal is to learn the general characteristics of detecting entities in documents using a sufficiently large dataset; and, later, refine the acquired knowledge for particular tasks in this domain.

In order to train the first detection model of this approach, we have applied the two methods presented in the previous sections to the PubLayNet dataset, and also trained the models from-scratch. The results obtained for this dataset can be found in Table II, where we can notice that, for all the models, the traditional transfer learning approach achieves better results than the other two training methods. Hence, those models have been employed for the second step of this transfer learning approach.

III. RESULTS AND DISCUSSION

In this section, we present the results for our experiments that are summarised in Table III. From those results, we can draw some general conclusions about the architectures and different training approaches.

A.1. Best architecture: From the conducted experiments, we can conclude that the YOLO architecture consistently produces the best results independently of the training approach. In particular, the YOLO model improve the performance of the FasterRCNN models between a 10% and a 20%, except for the Marmot dataset; and, between a 5% and a 160% for EfficientDet.

A.2. From scratch: Due to the limited number of training examples in each dataset, this approach generally offers the lowest results, the exception is for the YOLO models. In

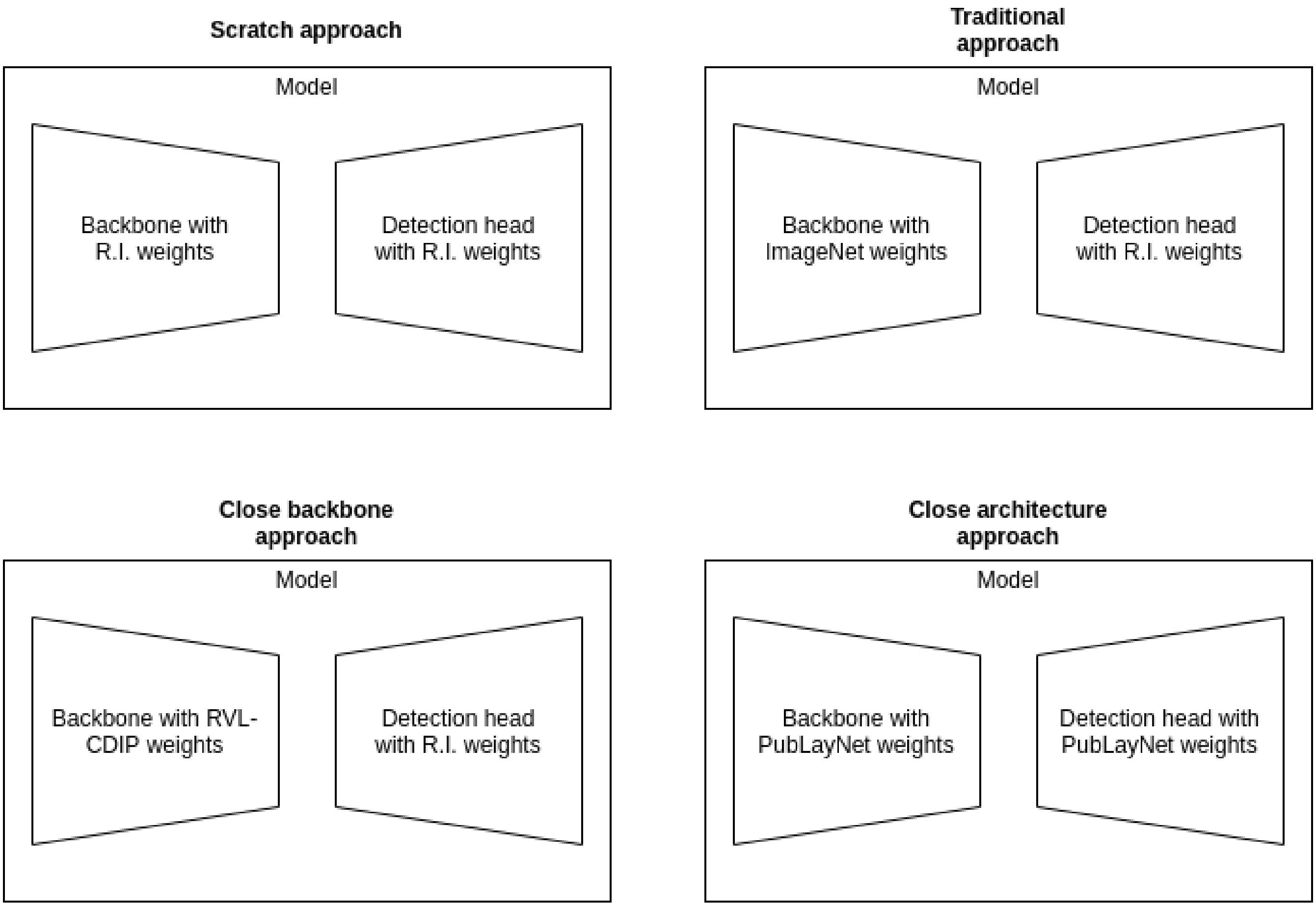

Fig. 2. Different training approaches employed in this work. R.I. stands for randomly initialised.

	From-scratch				Traditional				Close backbone			
Architecture	mAP	Prec	Rec	F1	mAP	Prec	Rec	F1	mAP	Prec	Rec	F1
FasterRCNN	0.90	0.97	0.98	0.98	**0.90**	0.97	0.99	0.98	0.90	0.97	0.98	0.97
EfficientDet	0.69	0.89	0.90	0.89	0.87	0.97	0.99	0.98	0.87	0.97	0.99	0.98
YOLO v4	0.90	0.98	0.96	0.97	0.90	0.98	0.96	0.97	0.88	0.97	0.95	0.96

TABLE II
MAP@[0.50:0.95:0.05], AND PRECISION, RECALL, AND F1-SCORE WITH IOU 0.5 ACHIEVED BY THE 3 STUDIED ARCHITECTURES BY TRAINING THEM FROM-SCRATCH, AND USING THE TRADITIONAL TRANSFER LEARNING APPROACH AND THE CLOSE BACKBONE TRANSFER APPROACH IN THE PUBLAYNET DATASET. BEST MODEL IS IN BOLD FACE.

addition, the training time and the number of epochs used in this approach is much higher than in the rest of approaches. This is due to the fact the rest of the approximations begin with a certain knowledge that makes them converge faster.

A.3. Traditional training: As it is known from the literature, the traditional approach using models pretrained on natural images improves the results obtained from the models initialised with random weights. This happens because the models take advantage of the learned characteristics in the largest classification datasets. This improvement varies from a 13% to a 90%.

A.4. Close backbone pretraining: The results obtained applying this approach depend on the architecture employed. Namely, all the FasterRCNN models trained using this approach obtain worse results than those obtained when training the models from scratch; on the contrary, the EfficientDet and YOLO models obtain better or similar results to those obtained with the traditional training process.

A.5. Close architecture fine-tuning: We can conclude that this approach produces the best results for all datasets and models. This show us, that the characteristics learned in the close domain are useful to obtain good results in the target task. We can also notice from the results of Table III that the datasets where this approach is more beneficial are those whose entities are a subset of those found in the PubLayNet dataset.

	Architecture	Scratch mAP	Scratch Prec	Scratch Rec	Scratch F1	Traditional mAP	Traditional Prec	Traditional Rec	Traditional F1	Close backbone mAP	Close backbone Prec	Close backbone Rec	Close backbone F1	Close architecture mAP	Close architecture Prec	Close architecture Rec	Close architecture F1
ICDAR	FasterRCNN	0.51	0.79	0.87	0.83	0.67	0.86	0.90	0.88	0.46	0.77	0.89	0.82	0.72	0.90	0.93	0.91
	EfficientDet	0.45	0.65	0.88	0.75	0.54	0.73	0.94	0.82	0.54	0.73	0.94	0.82	0.63	0.81	0.96	0.88
	YOLO v4	0.79	0.93	0.91	0.92	0.77	0.92	0.87	0.90	0.78	0.92	0.90	0.91	**0.79**	0.93	0.91	0.92
FUNSD	FasterRCNN	0.11	0.26	0.45	0.33	0.22	0.42	0.60	0.50	0.05	0.14	0.32	0.20	0.25	0.46	0.65	0.54
	EfficientDet	0.06	0.14	0.36	0.20	0.09	0.21	0.44	0.28	0.09	0.22	0.44	0.30	0.12	0.26	0.49	0.34
	YOLO v4	0.29	0.66	0.67	0.67	0.27	0.68	0.69	0.69	0.32	0.71	0.72	0.72	**0.32**	0.68	0.7	0.69
UNLV	FasterRCNN	0.45	0.76	0.89	0.82	0.71	0.89	0.92	0.91	0.09	0.30	0.67	0.42	0.73	0.92	0.98	0.95
	EfficientDet	0.56	0.77	0.99	0.86	0.66	0.90	1	0.95	0.65	0.89	1	0.94	0.71	0.91	0.99	0.95
	YOLO v4	0.79	0.96	0.92	0.94	0.81	0.95	0.93	0.94	0.78	0.95	0.92	0.94	**0.81**	0.95	0.95	0.95
MARM.	FasterRCNN	0.02	0.07	0.52	0.12	0.76	0.96	1	0.98	0.02	0.07	0.34	0.12	**0.87**	0.96	0.98	0.97
	EfficientDet	0.64	0.82	0.97	0.89	0.77	0.91	0.99	0.95	0.76	0.91	1	0.95	0.80	0.91	1	0.95
	YOLO v4	0.83	0.95	0.93	0.94	0.83	0.94	0.96	0.95	0.82	0.94	0.93	0.94	0.84	0.94	0.94	0.94

TABLE III
MAP@[0.50 : 0.95 : 0.05] AND PRECISION, RECALL AND F1-SCORE WITH IOU 0.5 ACHIEVED BY THE FOUR ARCHITECTURES STUDIED USING THE THREE APPROACHES IN THE FOUR DIFFERENT DATASETS. BEST MODELS ARE IN BOLD FACE.

IV. CONCLUSIONS AND FURTHER WORK

In this work, we have studied different transfer learning approaches to build detection models for entity recognition on document images. In particular, we have examined 3 different transfer learning approaches with 3 different deep object detection architectures. The best results have been achieved with the close architecture transfer approach using the YOLO v4 architecture, except for the Marmot dataset, where the FasterRCNN model trained with the close architecture transfer approach achieved the best results. With this study, we can conclude that starting from a model trained in a large dataset, from a close domain, improves the performance of those models. In contrast, training a backbone of a detection model in a classification task, even in a close domain, does not have to improve the results. What is more, this can worse them. As further work, we plan to test other architectures and use other training techniques such as semi-supervised and self-supervised learning that can reduce the amount of annotated images required for training the deep learning models.

REFERENCES

[1] Asperti, A., Mastronardo, C.: The effectiveness of data augmentation for detection of gastrointestinal diseases from endoscopical images. In: 11th International Joint Conference on Biomedical Engineering Systems and Technologies. BIOSTEC 2019, vol. 2, pp. 199–205. SciTePress (2018)

[2] Bochkovskiy, A., Wang, C.Y., Liao, H.Y.M.: Yolov4: Optimal speed and accuracy of object detection. arXiv preprint arXiv:2004.10934 (2020)

[3] Borko, H., Bernick, M.: Automatic document classification. Journal of the ACM (JACM) **10**(2), 151–162 (1963)

[4] Bowen, G.A.: Document analysis as a qualitative research method. Qualitative research journal (2009)

[5] Brants, T., Chen, F., Tsochantaridis, I.: Topic-based document segmentation with probabilistic latent semantic analysis. In: Proceedings of the eleventh international conference on Information and knowledge management. pp. 211–218 (2002)

[6] Coffey, A.: Analysing documents. The SAGE handbook of qualitative data analysis pp. 367–380 (2014)

[7] Deng, J., et al.: ImageNet: A Large-Scale Hierarchical Image Database. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR09) (2009)

[8] Everingham, M., Gool, L., Williams, C.K., Winn, J., Zisserman, A.: The pascal visual object classes (voc) challenge. Int. J. Comput. Vision **88**(2), 303–338 (Jun 2010). https://doi.org/10.1007/s11263-009-0275-4, https://doi.org/10.1007/s11263-009-0275-4

[9] Fang, J., Tao, X., Tang, Z., Qiu, R., Liu, Y.: Dataset, ground-truth and performance metrics for table detection evaluation. In: 2012 10th IAPR International Workshop on Document Analysis Systems. pp. 445–449 (2012). https://doi.org/10.1109/DAS.2012.29

[10] Gao, L., Yi, X., Jiang, Z., Hao, L., Tang, Z.: Icdar2017 competition on page object detection. In: 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR). vol. 01, pp. 1417–1422 (2017). https://doi.org/10.1109/ICDAR.2017.231

[11] Gatos, B., Danatsas, D., Pratikakis, I., Perantonis, S.J.: Automatic table detection in document images. In: International Conference on Pattern Recognition and Image Analysis. pp. 609–618. Springer (2005)

[12] Gatos, B., Danatsas, D., Pratikakis, I., Perantonis, S.J.: Automatic table detection in document images. In: International Conference on Pattern Recognition and Image Analysis. pp. 609–618. Springer (2005)

[13] Guillaume Jaume, Hazim Kemal Ekenel, J.P.T.: Funsd: A dataset for form understanding in noisy scanned documents. In: Accepted to ICDAR-OST (2019)

[14] Harley, A.W., Ufkes, A., Derpanis, K.G.: Evaluation of deep convolutional nets for document image classification and retrieval. In: International Conference on Document Analysis and Recognition (ICDAR) (2015)

[15] He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)

[16] Hong, T., et al.: A pre-trained language model for understanding texts in document (2021), https://openreview.net/forum?id=punMXQEsPr0

[17] Howard, J., Gugger, S.: Fastai: A Layered API for Deep Learning. Information **11**, 108 (02 2020)

[18] Huang, Y., et al.: A YOLO-based Table Detection Method. In: 15th International Conference on Document Analysis and Recognition. ICDAR'19 (2019)

[19] Kerwat, M., George, R., Shujaee, K.: Detecting Knowledge Artifacts in Scientific Document Images - Comparing Deep Learning Architectures. In: 5th International Conference on Social Networks Analysis, Management and Security. pp. 147–152. SNAMS'18, IEEE (2018)

[20] Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) Computer Vision – ECCV 2014. pp. 740–755. Springer International Publishing, Cham (2014)

[21] Mensink, T., Uijlings, J., Kuznetsova, A., Gygli, M., Ferrari, V.: Factors of influence for transfer learning across diverse appearance domains and task types (2021)

[22] Mori, S., Suen, C.Y., Yamamoto, K.: Historical review of ocr research and development. Proceedings of the IEEE **80**(7), 1029–1058 (1992)

[23] Paszke, A., et al.: Pytorch: An imperative style, high-performance deep learning library. In: Advances in Neural Information Processing Systems 32, pp. 8024–8035. Curran Associates, Inc. (2019)

[24] Raghu, M., Zhang, C., Kleinberg, J., Bengio, S.: Transfusion: Understanding transfer learning for medical imaging. arXiv preprint arXiv:1902.07208 (2019)

[25] Razavian, A.S., Azizpour, H., Sullivan, J., et al.: CNN features off-the-shelf: An astounding baseline for recognition. In: IEEE Conference on Computer Vision and Pattern Recognition Workshops. pp. 512–519. CVPRW'14, IEEE (2014)

[26] Redmon, J.: Darknet: Open source neural networks in c. http://pjreddie.com/darknet/ (2013–2016)

[27] Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: towards real-time object detection with region proposal networks. IEEE transactions on pattern analysis and machine intelligence **39**(6), 1137–1149 (2016)

[28] Schreiber, S., Agne, S., Wolf, I., Dengel, A., Ahmed, S.: Deepdesrt: Deep learning for detection and structure recognition of tables in document images. In: 2017 14th IAPR international conference on document analysis and recognition (ICDAR). vol. 1, pp. 1162–1167. IEEE (2017)

[29] Shahab, A., Shafait, F., Kieninger, T., Dengel, A.: An open approach towards the benchmarking of table structure recognition systems. In: ACM International Conference Proceeding Series. pp. 113–120 (06 2010). https://doi.org/10.1145/1815330.1815345

[30] Simard, P., Steinkraus, D., Platt, J.C.: Best practices for convolutional neural networks applied to visual document analysis. In: 12th International Conference on Document Analysis and Recognition. vol. 2, pp. 958–964 (2003)

[31] Smith, L.: Cyclical learning rates for training neural networks. In: IEEE Winter Conference on Applications of Computer Vision. pp. 464–472 (2017)

[32] Tan, M., Pang, R., Le, Q.V.: Efficientdet: Scalable and efficient object detection. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 10778–10787 (2020). https://doi.org/10.1109/CVPR42600.2020.01079

[33] Tan, M., Le, Q.: Efficientnet: Rethinking model scaling for convolutional neural networks. In: International Conference on Machine Learning. pp. 6105–6114. PMLR (2019)

[34] Xu, Y., Li, M., Cui, L., Huang, S., Wei, F., Zhou, M.: Layoutlm: Pre-training of text and layout for document image understanding. In: Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD '20), pp. 1192-1200. ACM (2020). https://doi.org/10.1145/3394486.3403172

[35] Yi, J., Nasukawa, T., Bunescu, R., Niblack, W.: Sentiment analyzer: Extracting sentiments about a given topic using natural language processing techniques. In: Third IEEE international conference on data mining. pp. 427–434. IEEE (2003)

[36] Zhong, X., Tang, J., Yepes, A.J.: Publaynet: largest dataset ever for document layout analysis. arXiv preprint arXiv:1908.07836 (2019)