REFERENCES

Farahani, A., Voghoei, S., Rasheed, K., & Arabnia, H. R. (2020). A brief review of domain adaptation. *arXiv preprint arXiv:2010.03978*. https://doi.org/10.48550/arXiv.2010.03978

Ganin, Y., & Lempitsky, V. (2015). Unsupervised domain adaptation by backpropagation. In *Proceedings of the 32nd International Conference on Machine Learning* (pp. 1180-1189). PMLR. https://proceedings.mlr.press/v37/ganin15.html

Hasani, M., & Khotanlou, H. (2019). An empirical study on position of the batch normalization layer in convolutional neural networks. In *2019 5th Iranian Conference on Signal Processing and Intelligent Systems (ICSPIS)* (pp. 1-4). IEEE. https://doi.org/10.1109/ICSPIS48872.2019.9066113

He, K., Zhang, X., Ren, S., & Sun, J. (2015). Deep residual learning for image recognition. *arXiv preprint arXiv:1512.03385*. https://doi.org/10.48550/arXiv.1512.03385

International Foundation for Electoral Systems. (2024). *2024 elections FAQ final*. https://www.ifes.org/sites/default/files/2024-02/2024_Elections_FAQ_Final.pdf

Ioffe, S., & Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*. https://doi.org/10.48550/arXiv.1502.03167

Jakarta Globe. (2024). Technical issues cause discrepancies in vote counting machine data. Retrieved April 10, 2025, from https://jakartaglobe.id/news/technical-issues-cause-discrepancies-in-vote-counting-machine-data

Javadecompilers.com. (2025). APK decompiler - decompile Android .apk. Retrieved April 13, 2025, from http://www.javadecompilers.com/apk

Kawal Pemilu. (2024). Real time election monitoring and data verification. Retrieved December 6, 2024, from https://kawalpemilu.org/

Koh, P. W., Sagawa, S., Marklund, H., Xie, S. M., Zhang, M., Balsubramani, A., Hu, W., Yasunaga, M., Phillips, R. L., Gao, I., Lee, T., David, E., Stavness, I., Guo, W., Earnshaw, B., Haque, I. S., Beery, S. M., Leskovec, J., Kundaje, A., ... Liang, P. (2021). WILDS: A benchmark of in-the-wild distribution shifts. In *Proceedings*

of the 38th International Conference on Machine Learning (pp. 5637-5664). PMLR. https://proceedings.mlr.press/v139/koh21a.html

Komisi Pemilihan Umum Republik Indonesia. (2025). *Pemilu 2024 - Hasil perhitungan suara*. Retrieved January 10, 2025, from https://pemilu2024.kpu.go.id/

Krause, A., Perona, P., & Gomes, R. (2010). Discriminative clustering by regularized information maximization. In *Advances in Neural Information Processing Systems*. Curran Associates, Inc. https://papers.nips.cc/paper_files/paper/2010/hash/42998cf32d552343bc8e460416382dca-Abstract.html

Kumari, T., Vardan, Y., Shambharkar, P. G., & Gandhi, Y. (2022). Comparative study on handwritten digit recognition classifier using CNN and machine learning algorithms. In *2022 6th International Conference on Computing Methodologies and Communication (ICCMC)* (pp. 882-888). IEEE. https://doi.org/10.1109/ICCMC53470.2022.9753756

LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE, 86*(11), 2278-2324. https://doi.org/10.1109/5.726791

Liang, J., He, R., & Tan, T. (2023). A comprehensive survey on test-time adaptation under distribution shifts. *arXiv preprint arXiv:2303.15361*. https://doi.org/10.48550/arXiv.2303.15361

Liang, J., Hu, D., & Feng, J. (2020). Do we really need to access the source data? Source hypothesis transfer for unsupervised domain adaptation. In *Proceedings of the 37th International Conference on Machine Learning* (pp. 6028-6039). PMLR. https://proceedings.mlr.press/v119/liang20a.html

Liu, C.-L., Nakashima, K., Sako, H., & Fujisawa, H. (2003). Handwritten digit recognition: Benchmarking of state-of-the-art techniques. *Pattern Recognition, 36*(10), 2271-2285. https://doi.org/10.1016/S0031-3203(03)00085-2

Mahkamah Konstitusi Republik Indonesia. (2024). IT expert reveals three sources of Sirekap issues. Retrieved April 10, 2025, from

https://en.mkri.id/news/details/2024-04-03/IT_Expert_Reveals_Three_Sources_of_Sirekap_Issues

McMahan, H. B., Moore, E., Ramage, D., Hampson, S., & y Arcas, B. A. (2023). Communication-efficient learning of deep networks from decentralized data. *arXiv preprint arXiv:1602.05629*. https://doi.org/10.48550/arXiv.1602.05629

Netron. (2025). *Netron*. Retrieved April 13, 2025, from https://netron.app/

Niu, S., Wu, J., Zhang, Y., Chen, Y., Zheng, S., Zhao, P., & Tan, M. (2023). Towards stable test-time adaptation in dynamic wild world. In *International Conference on Learning Representations*.

ONNX. (2025). ONNX | Home. Retrieved April 16, 2025, from https://onnx.ai/

Papers with Code. (2025a). MNIST benchmark (handwritten digit recognition). Retrieved April 10, 2025, from https://paperswithcode.com/sota/handwritten-digit-recognition-on-mnist

Papers with Code. (2025b). USPS benchmark (image clustering). Retrieved April 10, 2025, from https://paperswithcode.com/sota/image-clustering-on-usps

PyTorch. (2025). *PyTorch*. Retrieved April 16, 2025, from https://pytorch.org/

Ramazanova, M., Pardo, A., Ghanem, B., & Alfarra, M. (2025). Test-time adaptation for combating missing modalities in egocentric videos. *arXiv preprint arXiv:2404.15161*. https://doi.org/10.48550/arXiv.2404.15161

Salma. (2024). Indonesia's 2024 elections: Balancing technology and trust in digital age. *Universitas Gadjah Mada*. Retrieved April 10, 2025, from https://ugm.ac.id/en/news/indonesias-2024-elections-balancing-technology-and-trust-in-digital-age/

Simonyan, K., & Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*. https://doi.org/10.48550/arXiv.1409.1556

SIREKAP. (2025). Download SIREKAP 2024 2.48 Android APK file. *APKPure.com*. Retrieved April 13, 2025, from https://apkpure.com/sirekap-2024/id.go.kpu.sirekap2024/download/2.48

Sun, Y., Tzeng, E., Darrell, T., & Efros, A. A. (2019). Unsupervised domain adaptation through self-supervision. *arXiv preprint arXiv:1909.11825*. https://doi.org/10.48550/arXiv.1909.11825

Sun, Y., Wang, X., Liu, Z., Miller, J., Efros, A., & Hardt, M. (2020). Test-time training with self-supervision for generalization under distribution shifts. In *Proceedings of the 37th International Conference on Machine Learning* (pp. 9229-9248). PMLR. https://proceedings.mlr.press/v119/sun20b.html

Sun, Y., et al. (2025). Learning to (learn at test time): RNNs with expressive hidden states. *arXiv preprint arXiv:2407.04620*. https://doi.org/10.48550/arXiv.2407.04620

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., & Rabinovich, A. (2014). Going deeper with convolutions. *arXiv preprint arXiv:1409.4842*. https://doi.org/10.48550/arXiv.1409.4842

TensorFlow. (2025). *TensorFlow*. Retrieved April 16, 2025, from https://www.tensorflow.org/

tim-learn. (2025). tim-learn/SHOT: Code released for our ICML 2020 paper "Do we really need to access the source data? Source hypothesis transfer for unsupervised domain adaptation". Retrieved April 13, 2025, from https://github.com/tim-learn/SHOT/

Wang, D. (2025). *DequanWang/tent* [Computer software]. GitHub. https://github.com/DequanWang/tent

Wang, D., Shelhamer, E., Liu, S., Olshausen, B., & Darrell, T. (2020). Tent: Fully test-time adaptation by entropy minimization. In *International Conference on Learning Representations*. https://openreview.net/forum?id=uXl3bZLkr3c

Yohanes. (2025, February 23). Reverse engineering model handwritten digits recognition aplikasi mobile SIREKAP. *Amazing Grace*. Retrieved April 7, 2025, from https://blog.compactbyte.com/2024/02/23/reverse-engineering-model-handwritten-digits-recognition-aplikasi-mobile-sirekap/

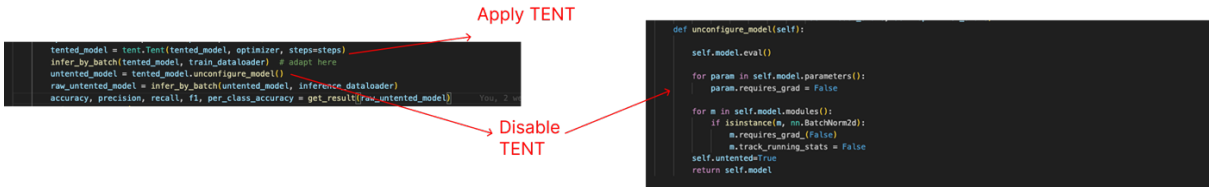
Zhao, H., Liu, Y., Alahi, A., & Lin, T. (2023). On pitfalls of test-time adaptation. In *Proceedings of the 40th International Conference on Machine Learning* (pp. 42058-42080). PMLR. https://proceedings.mlr.press/v202/zhao23d.html

APPENDIX

1. Post-processing 'applyHeuristic' code gained from decompilation using javadecompilers.n

```
/* JADX DEBUG: Multi-variable search result rejected for TypeSearchVarInfo{r2v8, resolved type: java.lang.Number} */
/* JADX WARNING: Multi-variable type inference failed */
/* Code decompiled incorrectly, please refer to instructions dump. */
private final int applyHeuristic(float[] r11) {
    /*
        r10 = this;
        kotlin.ranges.IntRange r0 = kotlin.collections.ArraysKt.getIndices((float[]) r11)
        java.lang.Iterable r0 = (java.lang.Iterable) r0
        java.util.Iterator r0 = r0.iterator()
        boolean r1 = r0.hasNext()
        if (r1 != 0) goto L_0x0012
        r0 = 0
        goto L_0x0043
    L_0x0012:
        java.lang.Object r1 = r0.next()
        boolean r2 = r0.hasNext()
        if (r2 != 0) goto L_0x001e
    L_0x001c:
        r0 = r1
        goto L_0x0043
    L_0x001e:
        r2 = r1
        java.lang.Number r2 = (java.lang.Number) r2
        int r2 = r2.intValue()
        r2 = r11[r2]
    L_0x0027:
        java.lang.Object r3 = r0.next()
        r4 = r3
        java.lang.Number r4 = (java.lang.Number) r4
        int r4 = r4.intValue()
        r4 = r11[r4]
        int r5 = java.lang.Float.compare(r2, r4)
        if (r5 >= 0) goto L_0x003c
        r1 = r3
        r2 = r4
    L_0x003c:
        boolean r3 = r0.hasNext()
        if (r3 != 0) goto L_0x0027
        goto L_0x001c
    L_0x0043:
        java.lang.Integer r0 = (java.lang.Integer) r0
        r1 = 0
        if (r0 == 0) goto L_0x004d
        int r0 = r0.intValue()
        goto L_0x004e
    L_0x004d:
        r0 = r1
    L_0x004e:
```

2. Code For TENT Evaluation in Adaptation Size Evaluation



3. Code For SHOT Evaluation in Adaptation Size Evaluation

```
if iter_num % interval_iter == 0 or iter_num == max_iter:
    netF.eval()
    netB.eval()
    acc, _ ,precision, recall, f1, class_accuracies= cal_acc_key_metrics(dset_loaders['test'], netF, netB, netC) # Real accuracy after running n times
    log_str = 'Task: {}, Iter:{}/{}; Accuracy = {:.2f}%'.format(args.dset, iter_num, max_iter, acc)
    saveResultTestAccuracy(acc, f1,precision,recall,args.iteration,args.dset_size,iter_num,max_iter,"test_accuracy_with_max_iter_iter_num_key_metrics_tracked","/workspace/Dwika/fyp/saved_results_with_key_metrics")
    columns = ['Class', 'Accuracy',"iter_num","max_iter"]

    for class_label, accuracy in class_accuracies.items():
        write_row_to_csv(f"/workspace/Dwika/fyp/saved_results_with_key_metrics/per_class_acc/{args.dset_size}_{args.iteration}.csv", columns, [class_label, accuracy,iter_num,max_iter])

    print(log_str+'\n')
    netF.train()
    netB.train()
```

4. **Translating Yohanes (2025) to English website (tested on Microsoft Edge Version 135.0.3179.73 (Official build) (arm64) Mac OS)**

Reverse Engineering Model Handwritten Digits Recognition Aplikasi Mobile SIREKAP

yohanes
23/02/2024
ai, android, reverse-

Di tulisan ini saya akan melakukan reverse engineering aplikasi SIREKAP 2024, mengekstrak model TensorFlow Lite untuk inference, lalu membuat aplikasi Android untuk mengetes model tersebut. Saya bandingkan juga dengan model sederhana yang jadi contoh tutorial tensorflow lite.

CARI KONTEN

ENHANCED BY Google

SEARCH

Blog ini bertenaga surya

PAGES

5. **Github Link: https://github.com/Dwikavindra/Final_Year (with details written in README.MD)**