

DAFTAR PUSTAKA

- Ajrana, A. (2022). *Rancang Bangun Aplikasi Mobile Klasifikasi Kanker Kulit Dengan Pemilihan Model Transfer Learning*. Universitas Hasanuddin.
- Alcalá-Rmz, V., Villagrana-Bañuelos, K. E., Celaya-Padilla, J. M., Galván-Tejada, J. I., Gamboa-Rosales, H., & Galván-Tejada, C. E. (2022). Convolutional neural network for monkeypox detection. *Proceedings of the International Conference on Ubiquitous Computing & Ambient Intelligence (UCAmI 2022)*, 89–100.
- Ali, S. N., Ahmed, M., Paul, J., Jahan, T., Sani, S. M., Noor, N., & Hasan, T. (2022). Monkeypox skin lesion detection using deep learning models: A feasibility study. *arXiv preprint arXiv:2207.03342*.
- Anggeriyane, E., Rahayu, S. F., Salamiah, D., Murizki, D. B., & Maulida, M. H. (2022). The Education on the Importance of Protecting from the Dangers of Chickenpox Through Audiovisual Learning Media: Edukasi Pentingnya Menjaga Diri dari Bahaya Cacar Air Melalui Media Pembelajaran Audiovisual. *Medani: Jurnal Pengabdian Masyarakat*, 1(3), 83–91.
- Baihaqi, K. A., & Zonyfar, C. (2022). Deteksi Lahan Pertanian Yang Terdampak Hama Tikus Menggunakan Yolo v5: Indonesia. *Syntax: Jurnal Informatika*, 11(02), 1–11.
- Budiyarto, L., Sabila, A. A., & Putri, H. C. (2023). Infeksi Cacar Monyet (Monkeypox). *Jurnal Medika Hutama*, 4(02 Januari), 3224–3236.
- Bunge, E. M., Hoet, B., Chen, L., Lienert, F., Weidenthaler, H., Baer, L. R., & Steffen, R. (2022). The changing epidemiology of human monkeypox—A potential threat? A systematic review. *PLoS neglected tropical diseases*, 16(2), e0010141.
- Chen, W., Sun, Q., Wang, J., Dong, J.-J., & Xu, C. (2018). A novel model based

on AdaBoost and deep CNN for vehicle classification. *Ieee Access*, 6, 60445–60455.

Effendi, D. W. (2022). *PERBANDINGAN OPTIMASI SGD DAN ADAM PADA ARSITEKTUR YOLOv5 (YOU ONLY LOOK ONCE) UNTUK DETEKSI ALAT PELINDUNG DIRI*. UPN Veteran Jawa Timur.

Gessain, A., Nakoune, E., & Yazdanpanah, Y. (2022). Monkeypox. *New England Journal of Medicine*, 387(19), 1783–1793.

HARIS, N. M. (2022). *IMPLEMENTASI ALGORITMA YOLOv7 PADA SISTEM PENGHITUNG BENIH KELAPA SAWIT SECARA REAL-TIME*.

Hubel, D. H., & Wiesel, T. N. (1962). Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *The Journal of physiology*, 160(1), 106.

Irfansyah, D., Mustikasari, M., & Suroso, A. (2021). Arsitektur Convolutional Neural Network (CNN) Alexnet Untuk Klasifikasi Hama Pada Citra Daun Tanaman Kopi. *Jurnal Informatika*, 6(2).

Jiang, L., Liu, H., Zhu, H., & Zhang, G. (2022). Improved YOLO v5 with balanced feature pyramid and attention module for traffic sign detection. *MATEC Web of Conferences*, 355, 3023.

Jubayer, F., Soeb, J. A., Mojumder, A. N., Paul, M. K., Barua, P., Kayshar, S., Akter, S. S., Rahman, M., & Islam, A. (2021). Detection of mold on the food surface using YOLOv5. *Current Research in Food Science*, 4, 724–728.

Kraemer, M. U. G., Tegally, H., Pigott, D. M., Dasgupta, A., Sheldon, J., Wilkinson, E., Schultheiss, M., Han, A., Oglia, M., & Marks, S. (2022). Tracking the 2022 monkeypox outbreak with epidemiological data in real-time. *The Lancet Infectious Diseases*, 22(7), 941–942.

Kumar, A. (2023). An XNOR-ResNet and spatial pyramid pooling-based YOLO

- v3-tiny algorithm for Monkeypox and similar skin disease detection. *The Imaging Science Journal*, 1–16.
- Li, H., Wang, P., You, M., & Shen, C. (2018). Reading car license plates using deep neural networks. *Image and Vision Computing*, 72, 14–23.
- Ludji, D. G., & Buan, F. C. H. (2022). Penerapan Metode Runge-Kutta Orde 4 pada Pemodelan Penularan Penyakit Cacar Monyet. *Jurnal Saintek Lahan Kering*, 5(2), 24–26.
- Marisah, M., Hilmi, I. L., & Salman, S. (2022). Studi dan tatalaksana terkait penyakit cacar monyet (monkeypox) yang menginfeksi manusia. *Jurnal Farmasetis*, 11(3), 201–208.
- Mukhtarom, R. (2020). *KAJIAN AKURASI MODEL ENSEMBLE CONVOLUTIONAL NEURAL NETWORK UNTUK KLASIFIKASI PENYAKIT TANAMAN TOMAT BERDASARKAN DATA CITRA DAUN*. Universitas Hasanuddin.
- MZ, A. R., Wijaya, I. G. P. S., & Bimantoro, F. (2020). Sistem pakar diagnosa penyakit kulit pada manusia dengan metode Dempster Shafer. *Journal of Computer Science and Informatics Engineering (J-Cosine)*, 4(2), 129–138.
- Nguyen, T.-K., Vu, L. T., Vu, V. Q., Hoang, T.-D., Liang, S.-H., & Tran, M.-Q. (2021). Analysis of Object Detection Models on Duckietown Robot Based on YOLOv5 Architectures. *International Journal of iRobotics*, 4(4), 17–22.
- O’Shea, K., & Nash, R. (2015). An introduction to convolutional neural networks. *arXiv preprint arXiv:1511.08458*.
- Olorunshola, O. E., Irhebhude, M. E., & Ewwiekpaefe, A. E. (2023). A Comparative Study of YOLOv5 and YOLOv7 Object Detection Algorithms. *Journal of Computing and Social Informatics*, 2(1), 1–12.
- Putra, W. S. E. (2016). Klasifikasi citra menggunakan convolutional neural

- network (CNN) pada caltech 101. *Jurnal Teknik ITS*, 5(1).
- Qelina, L., & Graharti, R. (2019). Human Monkeypox Virus: Respon Kesiapan Darurat Dunia. *Medula*, 9(03).
- Rahma, L., Syaputra, H., Mirza, A. H., & Purnamasari, S. D. (2021). Objek Deteksi Makanan Khas Palembang Menggunakan Algoritma YOLO (You Only Look Once). *Jurnal Nasional Ilmu Komputer*, 2(3), 213–232.
- Rizk, J. G., Lippi, G., Henry, B. M., Forthal, D. N., & Rizk, Y. (2022). Prevention and treatment of monkeypox. *Drugs*, 82(9), 957–963.
- Sabilla, I. A. (2020). Arsitektur Convolutional Neural Network (Cnn) Untuk Klasifikasi Jenis Dan Kesegaran Buah Pada Neraca Buah. *Institut Teknologi Sepuluh Nopember*.
- Sari, M., & Hairunisa, N. (2022). A REVIEW OF THE MONKEYPOX OUTBREAK IN INDONESIA IN 2022. *JURNAL KEDOKTERAN DIPONEGORO (DIPONEGORO MEDICAL JOURNAL)*, 11(5), 268–274.
- Setiawan, B. (2021). *PENERAPAN ALGORITMA YOU ONLY LOOK ONCE (YOLO) UNTUK DETEKSI TANAMAN MIANA BERBASIS ANDROID*. Universitas Muhammadiyah Ponorogo.
- Sihotang, W. D., Simbolon, C. C., Hartiny, J., Tindaon, D., & Sinaga, L. P. (2019). Analisis Kestabilan Model SEIR Penyebaran Penyakit Campak dengan Pengaruh Imunisasi dan Vaksin MR. *Jurnal Matematika, Statistika dan Komputasi*, 16(1), 107–113.
- Sitaula, C., & Shahi, T. B. (2022). Monkeypox virus detection using pre-trained deep learning-based approaches. *Journal of Medical Systems*, 46(11), 78.
- Suhendro, J. M., Sudarma, M., & Khrisne, D. C. (2021). Rancang Bangun Aplikasi Seluler Penyedia Jasa Perawatan dan Kecantikan Menggunakan Framework Flutter. *Jurnal SPEKTRUM*, 8.

- Surbakti, A. W. A., & Putri, R. E. (2022). Penghitung Pengunjung dan Deteksi Masker Menggunakan OpenCV dan YOLO. *CHIPSET*, 3(02), 83–93.
- Susanto, N. A. (2022). *ASUHAN KEPERAWATAN PADA An. M DENGAN VARICELLA (CHICKENPOX) ATAU CACAR AIR Di Ruang Rawat Inap C-Rs Wawa Husada Kepanjen*. Universitas Muhammadiyah Malang.
- Wang, C.-Y., Bochkovskiy, A., & Liao, H.-Y. M. (2022). YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. *arXiv preprint arXiv:2207.02696*.
- Widyaya, J. E., & Budi, S. (2021). Pengaruh Preprocessing Terhadap Klasifikasi Diabetic Retinopathy dengan Pendekatan Transfer Learning Convolutional Neural Network. *Jurnal Teknik Informatika Dan Sistem Informasi*, 7(1).
- Wulandari, P. (2019). *Klasifikasi Tingkat Keganasan Kanker Serviks Menggunakan Metode Deep Residual Network (ResNet)*. UIN Sunan Ampel Surabaya.
- Zahtamal, Z., Restuastuti, T., Restila, R., & Yusdiana, Y. (2020). Pengelolaan kesehatan kulit masyarakat pada masa pandemi Covid-19 di Desa Ranah Kecamatan Kampar. *Unri Conference Series: Community Engagement*, 2, 246–254.

LAMPIRAN

Lampiran 1. Kode Program

Source Code Model

Berikut merupakan *source code* yang digunakan pada pengimplementasian algoritma *YOLO*

Import Library

```
import random
import zipfile,os
from PIL import Image, ImageDraw
import glob
import numpy as np
import matplotlib.pyplot as plt
import shutil
```

Directory Dataset

```
local_zip = '/content/drive/MyDrive/skindiseasey.zip'
zip_ref = zipfile.ZipFile(local_zip, 'r')
zip_ref.extractall('/tmp')
zip_ref.close()
```

Inisiasi Directory Data train, test dan val

```
root_dir = os.path.join('/tmp', 'labels')
train_dir = os.path.join(root_dir , 'train')
test_dir = os.path.join(root_dir , 'test')
val_dir = os.path.join(root_dir , 'val')
```

Mapping nama class

```
class_name_to_id_mapping = {"Chickenpox": 0,
                             "Measles": 1,
                             "Monkeypox": 2,
                             "Normal": 3}
```

Memisahkan file anotasi dan gambar

```
annotations_train = [os.path.join(train_dir, x) for x in
os.listdir(train_dir) if x[-3:] == ".txt"]
annotations_val = [os.path.join(val_dir, x) for x in
os.listdir(val_dir) if x[-3:] == ".txt"]
annotations_test = [os.path.join(test_dir, x) for x in
os.listdir(test_dir) if x[-3:] == ".txt"]

images_train = [os.path.join(train_dir, x) for x in
os.listdir(train_dir) if '.jpg' in x.lower()]
images_val = [os.path.join(val_dir, x) for x in
os.listdir(val_dir) if '.jpg' in x.lower()]
images_test = [os.path.join(test_dir, x) for x in
os.listdir(test_dir) if '.jpg' in x.lower()]
```

Mencoba anotasi pada gambar

```
random.seed(88)

class_id_to_name_mapping =
dict(zip(class_name_to_id_mapping.values(),
class_name_to_id_mapping.keys()))

def plot_bounding_box(image, annotation_list):
    annotations = np.array(annotation_list)
    w, h = image.size

    plotted_image = ImageDraw.Draw(image)

    transformed_annotations = np.copy(annotations)
    transformed_annotations[:, [1, 3]] = annotations[:, [1, 3]] *
w
    transformed_annotations[:, [2, 4]] = annotations[:, [2, 4]] *
h

    transformed_annotations[:, 1] =
transformed_annotations[:, 1] - (transformed_annotations[:, 3] /
2)
    transformed_annotations[:, 2] =
transformed_annotations[:, 2] - (transformed_annotations[:, 4] /
2)
    transformed_annotations[:, 3] =
transformed_annotations[:, 1] + transformed_annotations[:, 3]
```

```

transformed_annotatons[:,4] =
transformed_annotatons[:,2] + transformed_annotatons[:,4]

for ann in transformed_annotatons:
    obj_cls, x0, y0, x1, y1 = ann
    plotted_image.rectangle((x0,y0), (x1,y1))

    plotted_image.text((x0, y0 - 10),
class_id_to_name_mapping[(int(obj_cls))])

plt.imshow(np.array(image))
plt.show()

# Get any random annotation file
annotation_file = random.choice(annotations_train)
with open(annotation_file, "r") as file:
    annotation_list = file.read().split("\n")[:-1]
    annotation_list = [x.split(" ") for x in annotation_list]
    annotation_list = [[float(y) for y in x ] for x in
annotation_list]

#Get the corresponding image file
image_file = annotation_file.replace("txt", "jpg")

#Load the image
image = Image.open(image_file)

#Plot the Bounding Box
plot_bounding_box(image, annotation_list)

```

Membuat folder anotasi dan gambar

```

!mkdir /tmp/labels/train/labels /tmp/labels/val/labels
/tmp/labels/test/labels
!mkdir /tmp/labels/train/images /tmp/labels/val/images
/tmp/labels/test/images

```


Pindahkan file anotasi dan gambar ke dalam masing – masing folder

```
def move_files_to_folder(list_of_files, destination_folder):
    for f in list_of_files:
        shutil.move(f, destination_folder)

# Move the splits into their folders
move_files_to_folder(images_train, '/tmp/labels/train/images')
move_files_to_folder(images_val, '/tmp/labels/val/images')
move_files_to_folder(images_test, '/tmp/labels/test/images')
move_files_to_folder(annotations_train,
'/tmp/labels/train/labels')
move_files_to_folder(annotations_val,
'/tmp/labels/val/labels')
move_files_to_folder(annotations_test,
'/tmp/labels/test/labels')
```

Melakukan *cloning* di repository *YOLO*

```
!git clone https://github.com/WongKinYiu/yolov7.git
```

Menginstall *requirements* yang diperlukan di *YOLO*

```
%cd /content/yolov7
!pip install -qr requirements.txt
```

Melakukan *training* pada model

```
!python train.py --img 640 --cfg cfg/training/yolov7.yaml --
hyp hyp.scratch.custom.yaml --batch 4 --epochs 100 --data
monkeypox.yaml --weights /content/drive/MyDrive/best.pt --
workers 8 --name yolo_monkey_det
```

Melakukan *testing* pada model

```
!python detect.py --source /tmp/labels/test/images/ --weights
runs/train/yolo_monkey_det/weights/best.pt --conf 0.25 --name
yolo_monkey_det
```

Source Code Mobile Apps

Berikut *source code mobile apps* yang dapat diakses melalui *qr-code*

