

Kamu telah **[Menerapkan Seluruh Kriteria Utama]** pada submission ini, diantaranya yaitu:

- Bebas Memilih Dataset yang Ingin Dipakai, tetapi Harus Memiliki Minimal 1000 Gambar ✓
- Tidak Diperbolehkan Menggunakan Dataset yang Sudah Pernah Digunakan ✓
- Dataset Dibagi Menjadi Train Set, Test Set dan Validation Set. ✓
- Model Harus Menggunakan Model Sequential, Conv2D, Pooling Layer ✓
- Akurasi pada Training dan Testing Set Minimal Sebesar 85% ✓
- Membuat Plot Terhadap Akurasi dan Loss Model ✓
- Menyimpan Model ke Dalam Format SavedModel, TF-Lite dan TFJS ✓

### **Additional Review**

Selain itu, kamu juga sudah **[Menerapkan Kriteria Tambahan]**, seperti:

- Mengimplementasikan Callback ✓
- Gambar-gambar pada dataset asli memiliki resolusi yang tidak seragam (Tanpa preprocessing) ✓
- Dataset yang digunakan berisi minimal 10000 gambar. ✓
- Akurasi pada training set dan testing set minimal 95%. ✓
- Memiliki 3 buah kelas atau lebih. ✓
- Melakukan inference menggunakan salah satu model (TF-Lite, TFJS atau savedmodel). ✓

### **Saran**

Untuk meningkatkan pemahaman dan skill kamu, berikut beberapa saran yang bisa kamu ikuti:

#### **Saran [Kriteria Tambahan - 4]**

- **Akurasi pada training set dan testing set minimal 95% (Pastikan tidak overfit/underfit)**
  - Lakukan tuning hyperparameter secara mendalam untuk mencapai akurasi yang lebih tinggi.
  - Eksplorasi penggunaan arsitektur model yang lebih kompleks atau transfer learning untuk meningkatkan performa.
  - Terapkan teknik validasi silang (cross-validation) untuk memastikan model tidak overfitting.

## Saran [Peningkatan Proyek]

- **Integrasi Transfer Learning**

Terapkan model pra-latih (misal: **ResNet**, **EfficientNet**, **MobileNet**) sebagai backbone, kemudian fine-tune pada dataset Anda. Transfer learning seringkali mempercepat konvergensi dan meningkatkan akurasi, terutama jika data asli terbatas.

- **Advanced Data Augmentation**

Perluas teknik augmentasi dengan **mixup**, **CutMix**, atau **random erasing** selain rotasi dan flip biasa. Ini dapat **meningkatkan generalisasi model** dengan menambahkan keragaman yang lebih realistis pada data pelatihan.

- **Hyperparameter Optimization Otomatis**

Gunakan library seperti **Optuna** atau **Keras Tuner** untuk **menjelajah ruang hyperparameter** secara otomatis (learning rate, batch size, arsitektur layer, dsb.). Hasilnya lebih konsisten dan seringkali lebih optimal daripada tuning manual.

- **Dokumentasi yang Komprehensif**

Pastikan setiap bagian dari kode, mulai dari preprocessing, pemodelan, hingga evaluasi, diberi komentar dan dokumentasi yang jelas. Buatlah README yang detail untuk memudahkan orang lain (atau dirimu sendiri di masa depan) dalam memahami alur dan tujuan dari proyek yang kamu kerjakan.

- **Evaluasi Kinerja dan Error Analysis**

Lakukan analisis error secara mendalam pada hasil prediksi model. Identifikasi pola kesalahan yang sering terjadi dan cari solusi yang tepat, baik dengan memperbaiki preprocessing, menambah data, atau menyempurnakan arsitektur model.

Dengan mengikuti saran-saran tersebut, kamu akan semakin mengasah keterampilan dan pengetahuan dalam pengembangan proyek [Klasifikasi Gambar](#). Terus semangat untuk menjadi **Machine Learning Engineer** dan jangan ragu untuk bereksperimen!

**Overall, kamu sudah mengerjakan submission ini dengan sangat baik! Well done!**