

Здравствуйте! Далее комментарии по коду работы:

```
PATH = "../working/car/" # рабочая директория
```

Если вы работаете на Kaggle, то можно использовать директорию /Kaggle/temp, как показано в бейзлайне, оттуда файлы автоматически удаляются после завершения работы ноутбука, подробнее см. [здесь](#).

```
augmentations.Blur(blur_limit=1, p=0.2),
    augmentations.ElasticTransform(alpha=0.1, sigma=5, alpha_affine=2,
                                   border_mode=cv2.BORDER_CONSTANT, value =
[0,0,0], #fill emptiness with (0,0,0)
    ),
```

Проблема в том, что такие аугментации могут очень долго выполняться, но экспериментально замечено, что с них не так много пользы. Обычно достаточно просто случайного смещения, поворота и отражения по горизонтали. Если у вас долго выполняются аугментации, то сам процесс может сильно замедляться.

```
test_datagen =
ImageDataAugmentor(augment=augmentations.ToFloat(max_value=255), seed=SEED)
test_generator = test_datagen.flow_from_directory(PATH+'train/',
                                                  batch_size=BATCH_SIZE,
                                                  shuffle=True,
                                                  )
```

Совершенно правильно, что вы делаете валидацию без аугментаций изображений

```
checkpoint = ModelCheckpoint('best_model.hdf5' , monitor = ['val_accuracy'] ,
verbose = 1 , mode = 'max')
```

Здесь нужно добавить параметр save_best_only=True

```
model.fit_generator(
```

Это рабочий, но устаревший метод, который сейчас полностью эквивалентен fit

Виджу, что у вас обучение выполняется в 3 шага с помлойной разморозкой. Я бы рекомендовал сначала попробовать обучить всю модель целиком, вдруг результат будет не хуже, а то и лучше? И код получится проще.

```
predictions = []
for _ in range(10):
    predictions.append(model.predict(test_sub_generator, verbose=1))
    test_sub_generator.reset()
predictions = np.array(predictions)
```

У вас test_sub_generator – это генератор изображений без аугментаций, поэтому будут выданы одни и те же предсказания на каждом из 10 шагов. Для ТТА нужно сделать генератор с аугментациями.

Отзыв подготовил ментор проекта Олег Седухин. Если есть вопросы, можете задать их в канале #0-project_7-ford_vs_ferrari, постараюсь помочь разобраться. Успехов в дальнейшем обучении!