



UNIVERSITY
OF ARTIFICIAL
INTELLIGENCE

Герасимов Денис Георгиевич

Классификация деталей изделий машиностроения по чертежам

31.08.2021

Дипломный
руководитель
Дмитрий Носков

Ноябрь 2020



Цель

- Создание информационной системы способной по чертежу детали относить ее к определенному классу по типу обработки, заготовки или др.

Задачи

- Собрать базу изображений чертежей деталей
- Преобразовать в формат PNG
- Удалить лишние данные(сокращение формата ИД),
бинаризация изображений
- Разбить на классы, создать обучающую и проверочную выборки
- Создать архитектуру НС
- Тестирование работы НС, графическое представление ошибок
- Оптимизация параметров сети для получения требуемой точности



Обучающая база

Объем базы

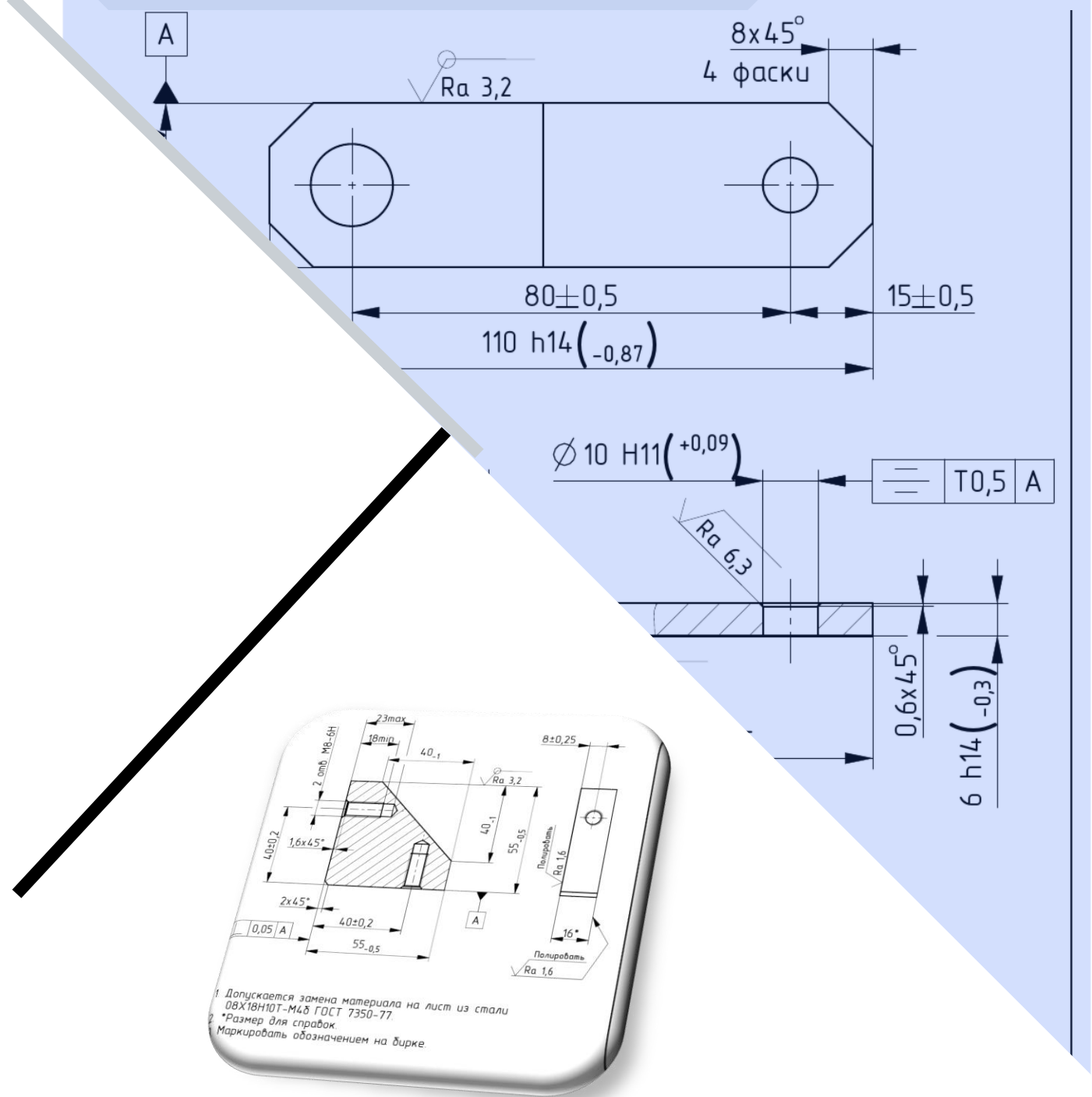
3200 чертежей формата A4 выполненных при помощи CAD

Источник сбора данных

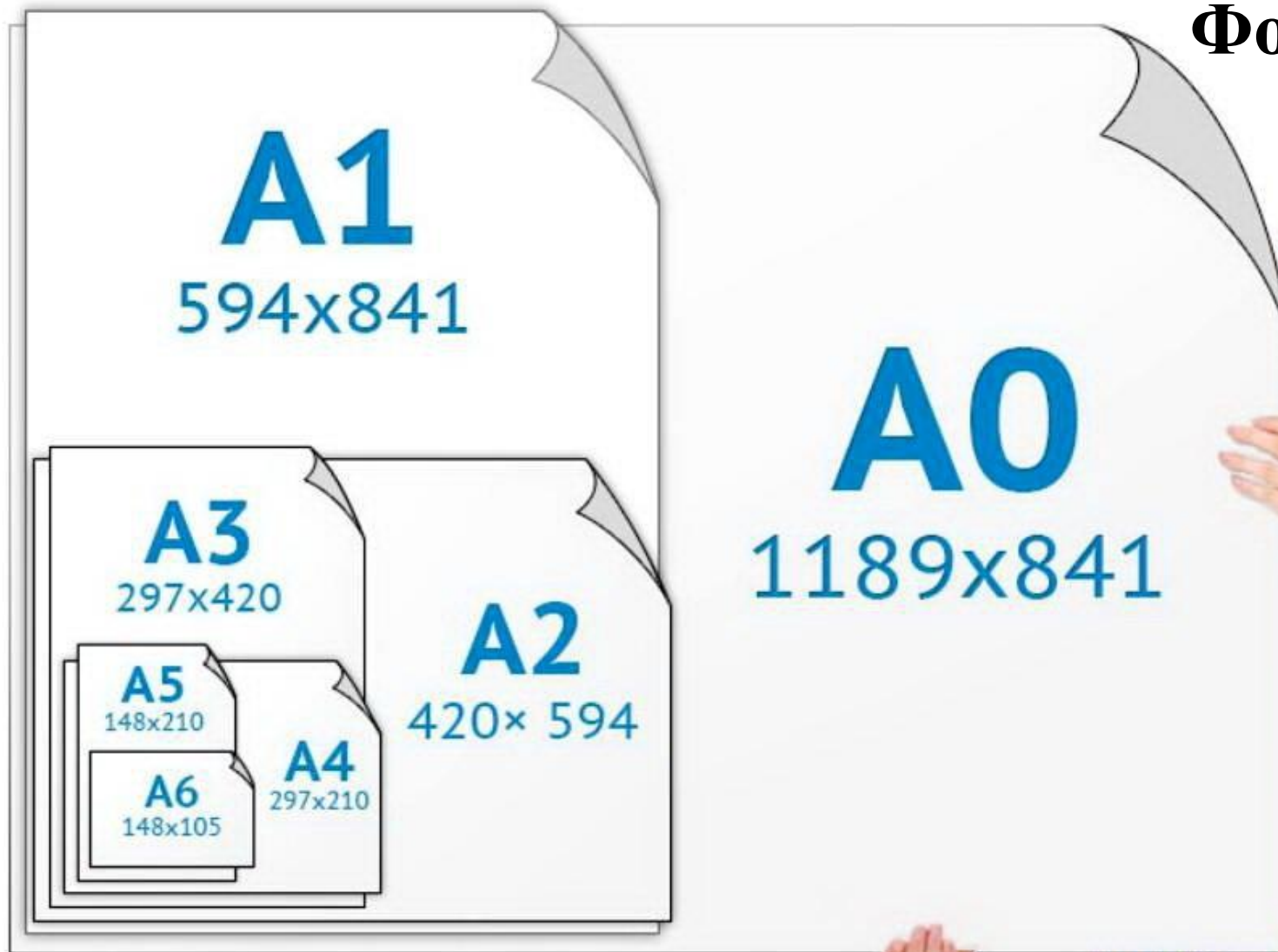
Личный архив, работа со специалистами

Возникшие трудности при сборе базы

- Преобладание деталей нескольких классов
- Низкое качество изображений
- Разный формат чертежей A4, A3, A2, A1, A4x3....
- Является собственностью фирм, а не отдельных лиц



Форматы чертежей

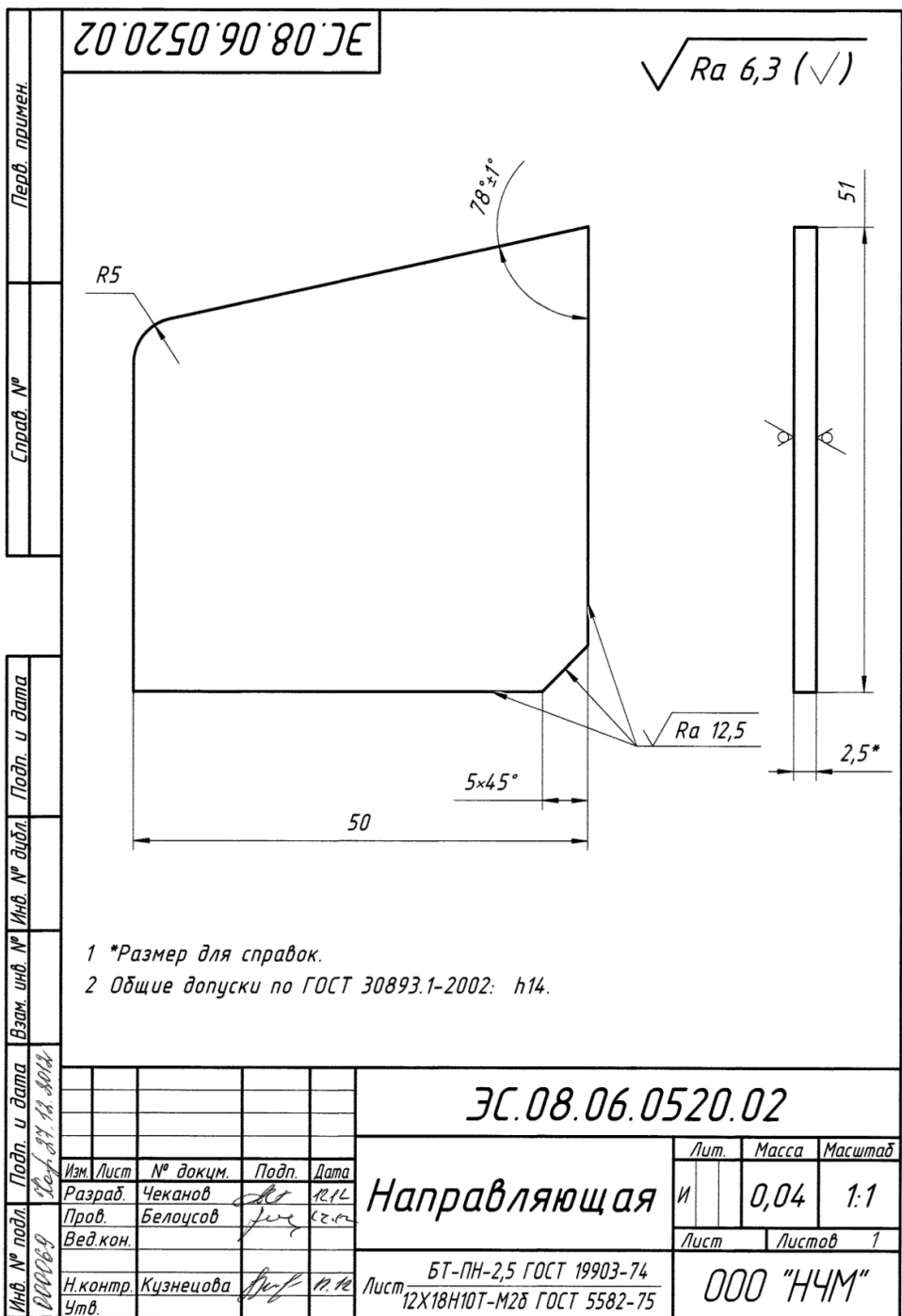
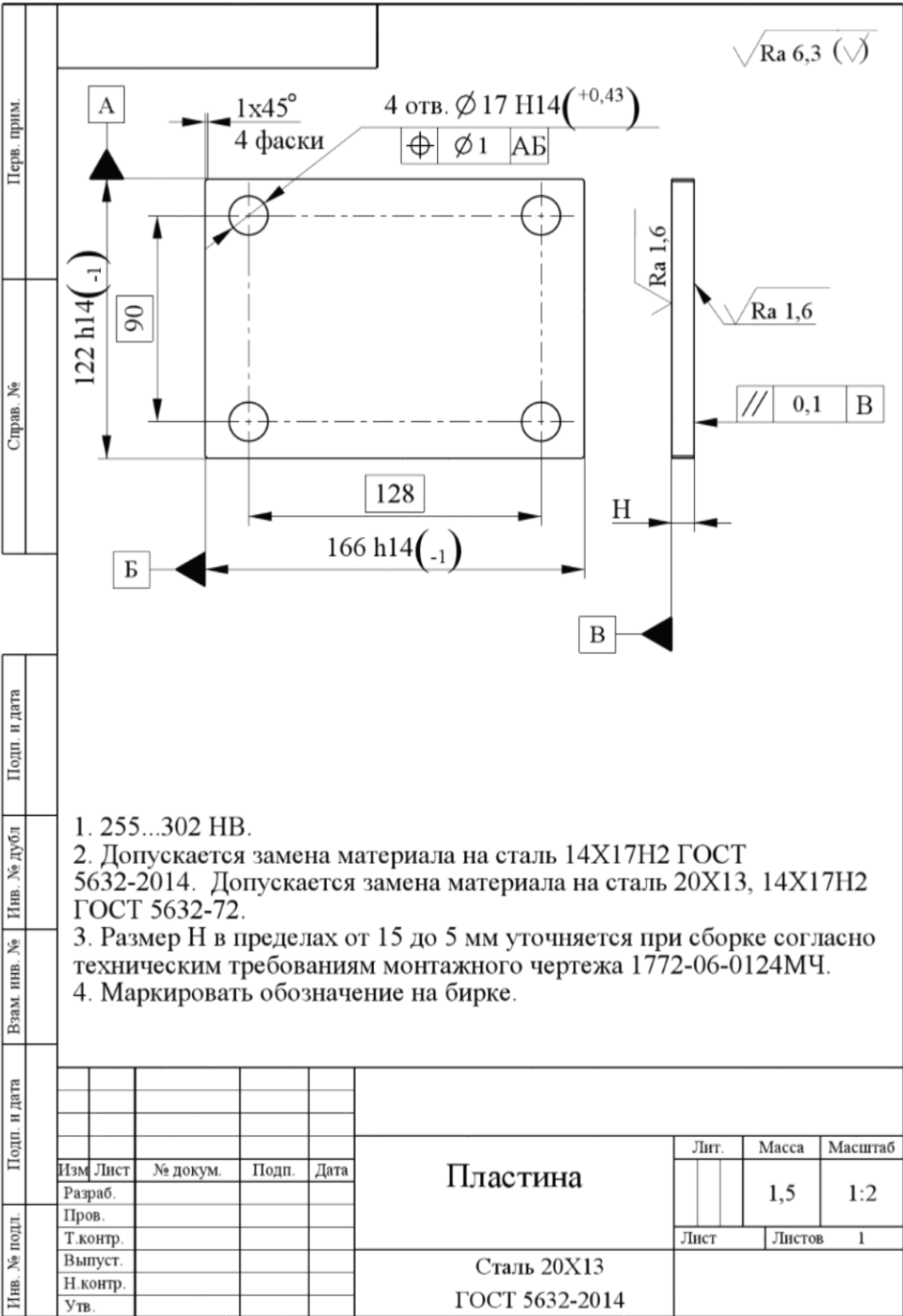


	75 dpi	150 dpi	300 dpi
A4 в px	877x620	1754x1240	3508x2480
A3 в px	877x1240	1754x2480	3508x4961

Примеры
чертежей
деталей.
Формат А4.

Изображение

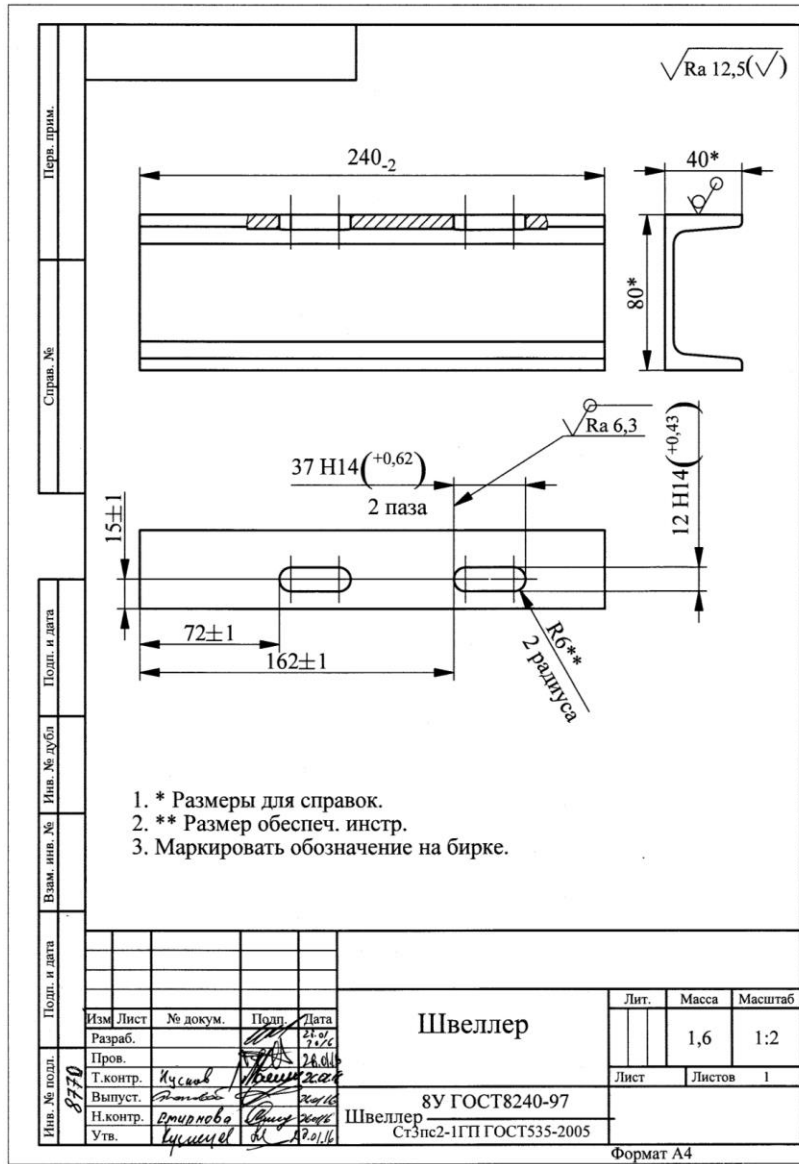
Разрешение	1240 x 1754
Ширина	1240 пикселей
Высота	1754 пикселей



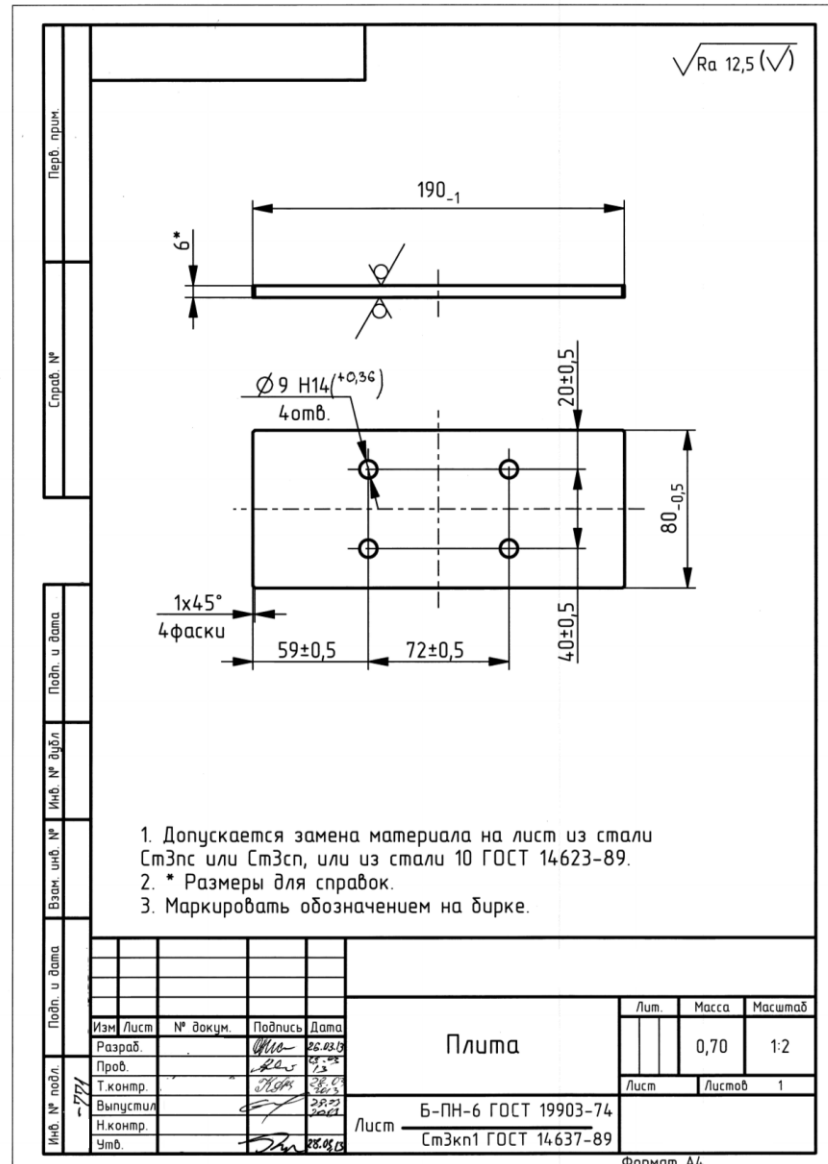
Классы деталей



Тело вращения

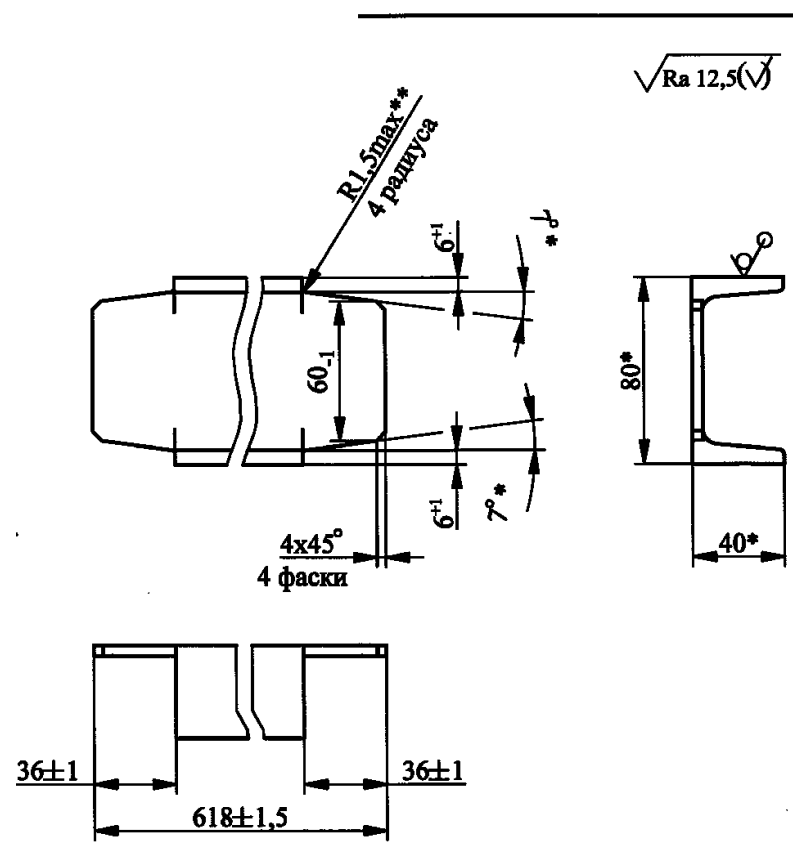
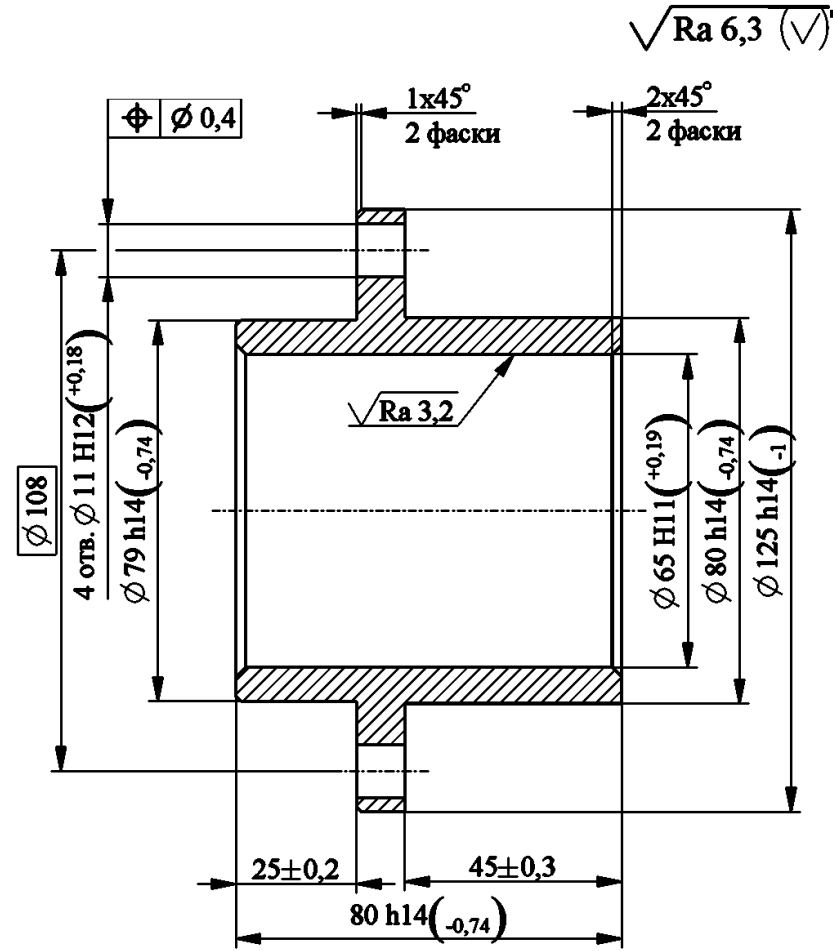


Прокат(швеллер)

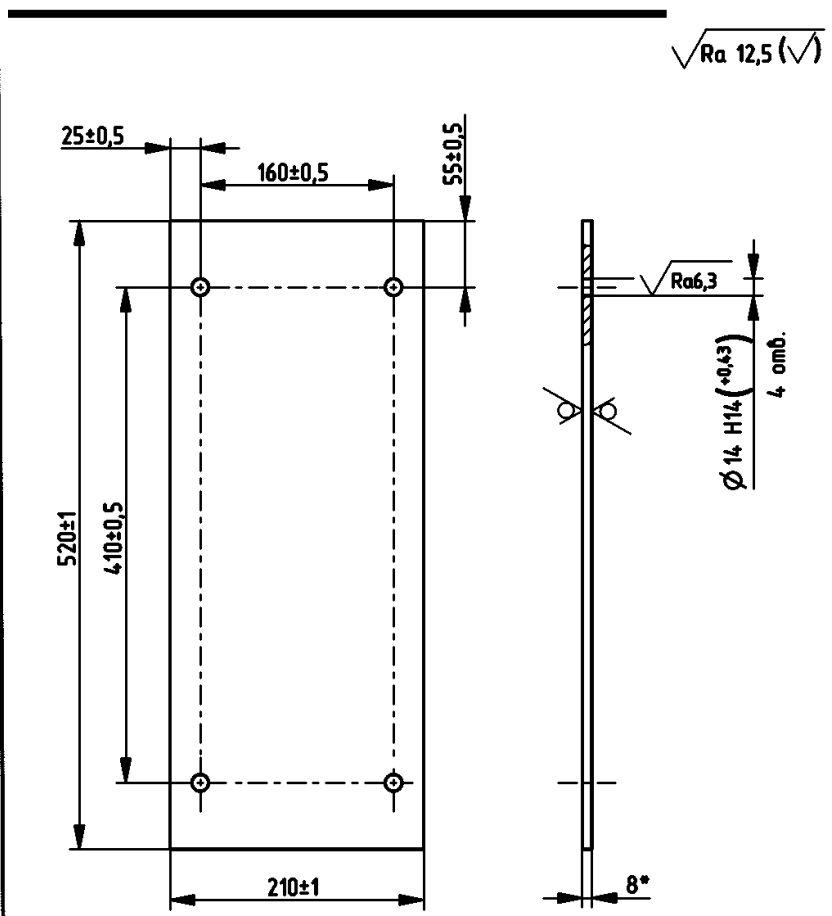


Лист

Формат исходных данных



1. * Размеры для справок.
2. ** Размер обеспеч. инстр.
3. Маркировать обозначением на бирке.



Допускается замена материала на лист из СтЗсп ГОСТ 14637-89 или стали 10 ГОСТ 1577-93.
*Размер для справок.
Маркировать обозначением на бирке.

Удаление рамок и штампов

Изображение

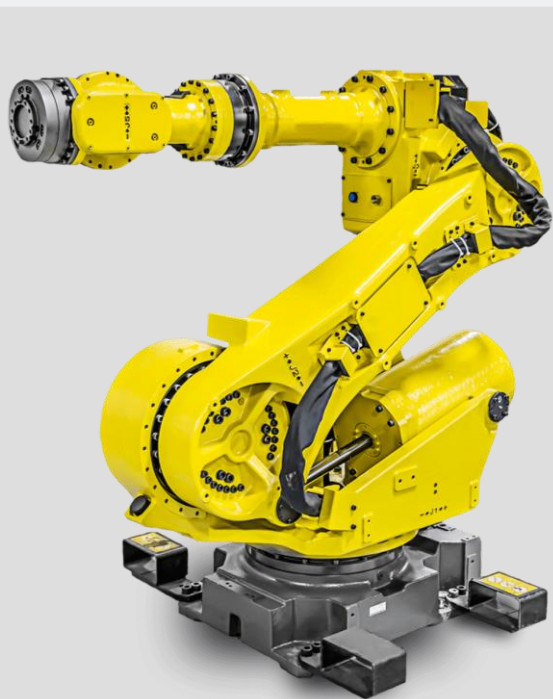
Разрешение	1040 x 1325
Ширина	1040 пикселей
Высота	1325 пикселей

План работы

- Определить влияние на точность работы НС:
 - Разрешение входного изображения
 - Размер выходного тензора (степень сжатия)
 - Размер окна свертки
 - Количество слоев сжатия
 - MaxPooling2D vs strides = 2
 - AveragePooling2D, GlobalAveragePooling2D, SeparableConv2D
- Первая модель. Последовательная сверточная нейронная сеть
- Модель сети с остаточными связями: повторное внедрение предыдущей исходящей информации
- Модель с несколькими параллельными ветвями
- Ассемблирование моделей

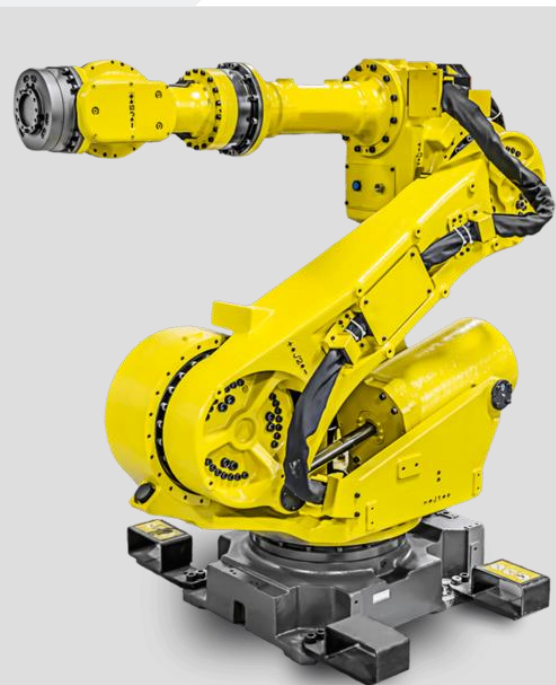


При работе с обычными изображениями при уменьшении размеров ключевая информация сохраняется



1200x1230

Исходное изображение



600x615

Уменьшить в 2 раза



300x307

Уменьшить в 4 раза

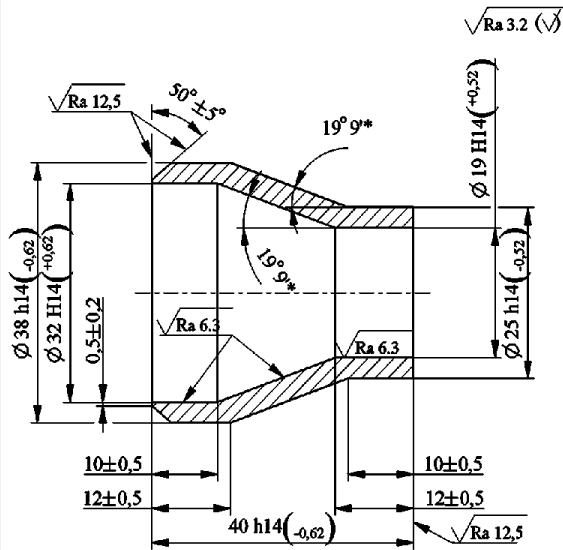


150x154

Уменьшить в 8 раз

Зависимость точности работы сети от разрешения исходного изображения (размер входного тензора)

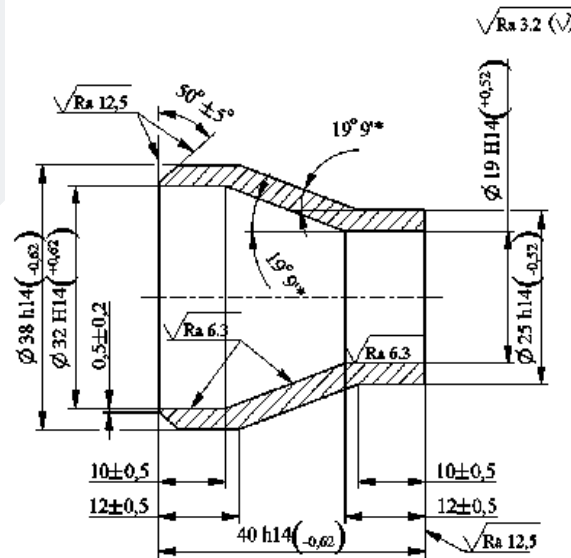
Точность работы сети зависит от разрешения исходного изображения



Допускается замена материала на сталь 08X18H10T ГОСТ 5632-201.
Допускается замена материала по ГОСТ 5632-72.
* Размеры для справок.
Маркировать обозначение на бирке.

1024x768

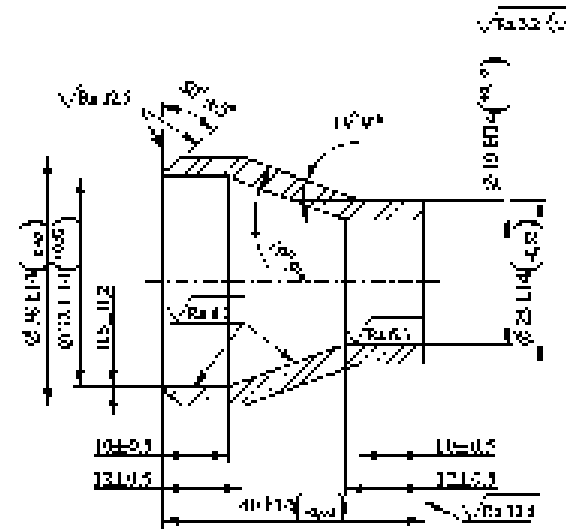
val_loss: 0.2724
val_accuracy: 0.9043



Допускается замена материала на сталь 08X18H10T ГОСТ 5632-201.
Допускается замена материала по ГОСТ 5632-72.
* Размеры для справок.
Маркировать обозначение на бирке.

512x384

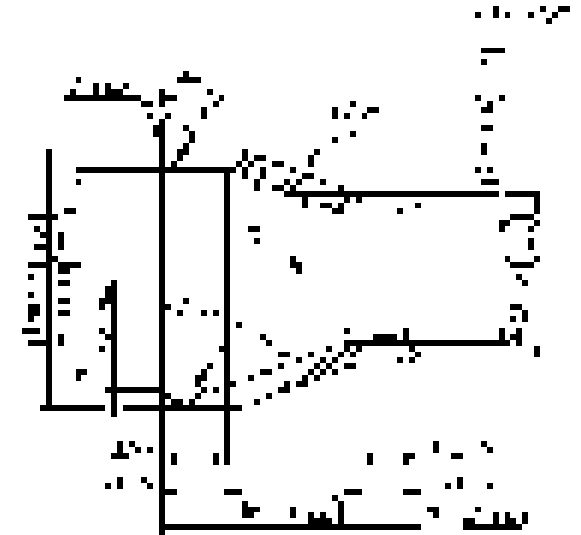
val_loss: 0.2843
val_accuracy: 0.9196



Допускается замена материала на сталь 08X18H10T ГОСТ 5632-201.
Допускается замена материала по ГОСТ 5632-72.
* Размеры для справок.
Маркировать обозначение на бирке.

256x192

val_loss: 0.4020
val_accuracy: 0.8435



Допускается замена материала на сталь 08X18H10T ГОСТ 5632-201.
Допускается замена материала по ГОСТ 5632-72.
* Размеры для справок.
Маркировать обозначение на бирке.

128x96

val_loss: 0.5045
val_accuracy: 0.8065

Зависимость точности работы сети от размера выходного тензора

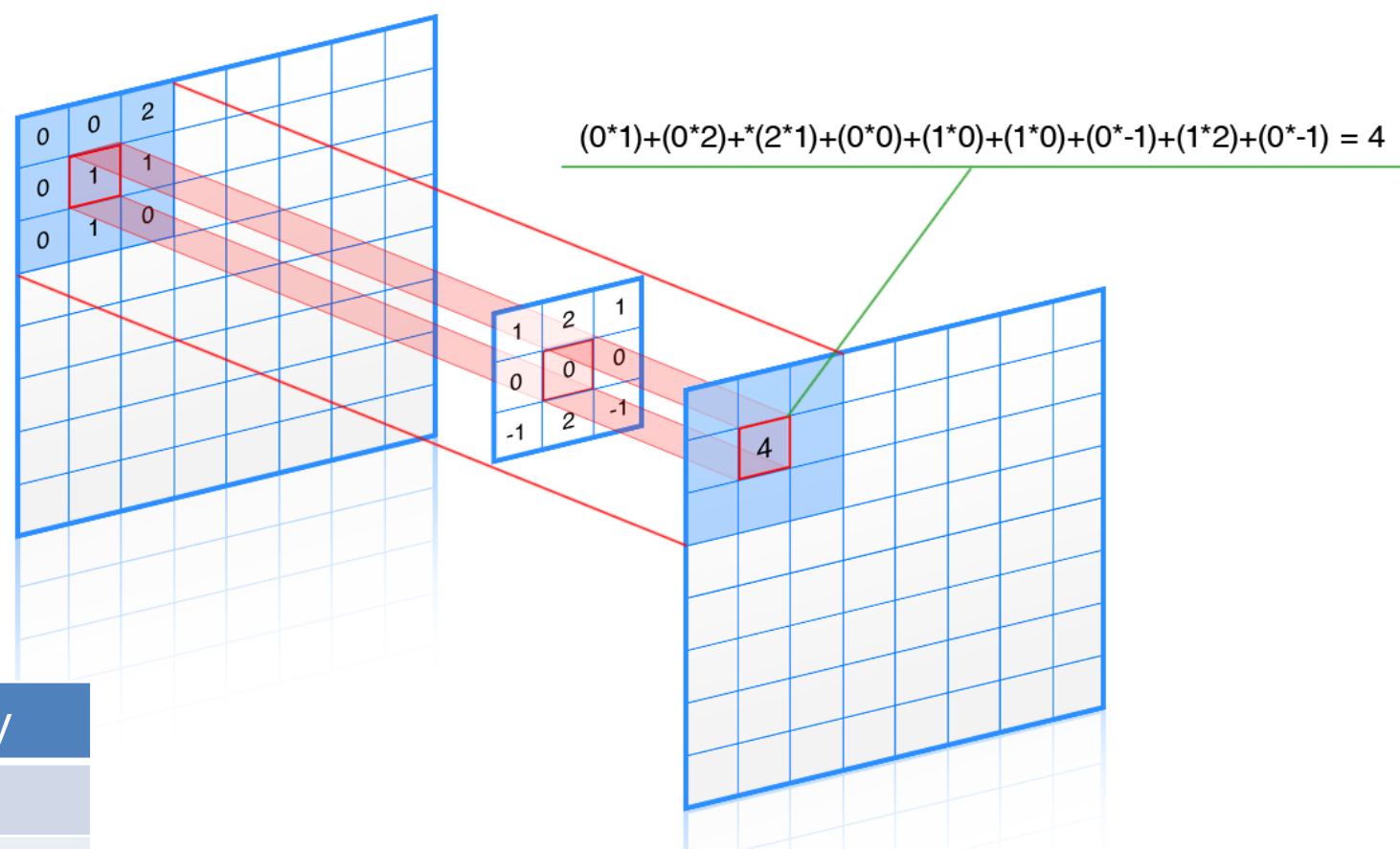
Точность работы сети зависит от размера выходного тензора

(None, 32, 24, 32)	(None, 16, 12, 128)	(None, 8, 6, 512)	(None, 4, 3, 512)
(None, 32, 24, 32)	(None, 16, 12, 128)	(None, 8, 6, 512)	(None, 4, 3, 512)
(None, 16, 12, 32)	(None, 8, 6, 128)	(None, 4, 3, 512)	(None, 2, 1, 1024)
(None, 16, 12, 32)	(None, 8, 6, 128)	(None, 4, 3, 512)	(None, 2, 1, 1024)
(None, 6144)	(None, 6144)	(None, 6144)	(None, 2048)
(None, 6144)	(None, 6144)	(None, 6144)	(None, 2048)
val_loss: 0.4179	val_loss: 0.2843	val_loss: 0.3049	val_loss: 0.3462
val_accuracy: 0.8435	val_accuracy: 0.9196	val_accuracy: 0.8804	val_accuracy: 0.8673

Зависимость точности работы сети от размера окна свертки

Выходной тензор : (None, 8, 6, 128)

	val_loss(min)	val_accuracy
2x2	0.3579	0.8696
3x3	0.2583	0.8978
4x4	0.3392	0.8565
5x5	0.3476	0.8674
6x6	0.3625	0.8522
7x7	0.9343	0.6087
8x8	1.0319	0.4391



Точность работы НС мало зависит от размера окна в диапазоне от 2x2 до 6x6.

Зависимость точности работы сети от количества слоев сжатия MaxPooling2D

Точность работы сети мало зависит от количества слоев сжатия

(None, 16, 12, 128)	(None, 32, 24, 128)	(None, 64, 48, 64)	(None, 512, 384, 16)
(None, 16, 12, 128)	(None, 32, 24, 128)	(None, 64, 48, 128)	(None, 512, 384, 128)
(None, 8, 6, 128)	(None, 8, 6, 128)	(None, 8, 6, 128)	(None, 8, 6, 128)
(None, 8, 6, 128)	(None, 8, 6, 128)	(None, 8, 6, 128)	(None, 8, 6, 128)
(None, 6144)	(None, 6144)	(None, 6144)	(None, 6144)
(None, 6144)	(None, 6144)	(None, 6144)	(None, 6144)
2x2x2x2x2x2	4x4x4	8x8	64
val_loss: 0.2843	val_loss: 0.2976	val_loss: 0.2325	val_loss: 0.2948
val_accuracy: 0.9196	val_accuracy: 0.9087	val_accuracy: 0.9239	val_accuracy: 0.8761

Зависимость точности работы сети от метода сжатия

MaxPooling2D

Strides=2

Strides=4

(None, 16, 12, 128)

(None, 16, 12, 128)

(None, 8, 6, 128)

(None, 8, 6, 128)

(None, 6144)

(None, 6144)

2x2x2x2x2x2

val_loss: 0.2843

val_accuracy: 0.9196

(None, 32, 24, 128)

(None, 32, 24, 128)

(None, 8, 6, 128)

(None, 8, 6, 128)

(None, 6144)

(None, 6144)

4x4x4

val_loss: 0.2976

val_accuracy: 0.9087

(None, 16, 12, 128)

(None, 16, 12, 128)

(None, 8, 6, 128)

(None, 8, 6, 128)

(None, 6144)

(None, 6144)

2x2x2x2x2x2

val_loss: 0.5084

val_accuracy: 0.8304

(None, 32, 24, 128)

(None, 32, 24, 128)

(None, 8, 6, 128)

(None, 8, 6, 128)

(None, 6144)

(None, 6144)

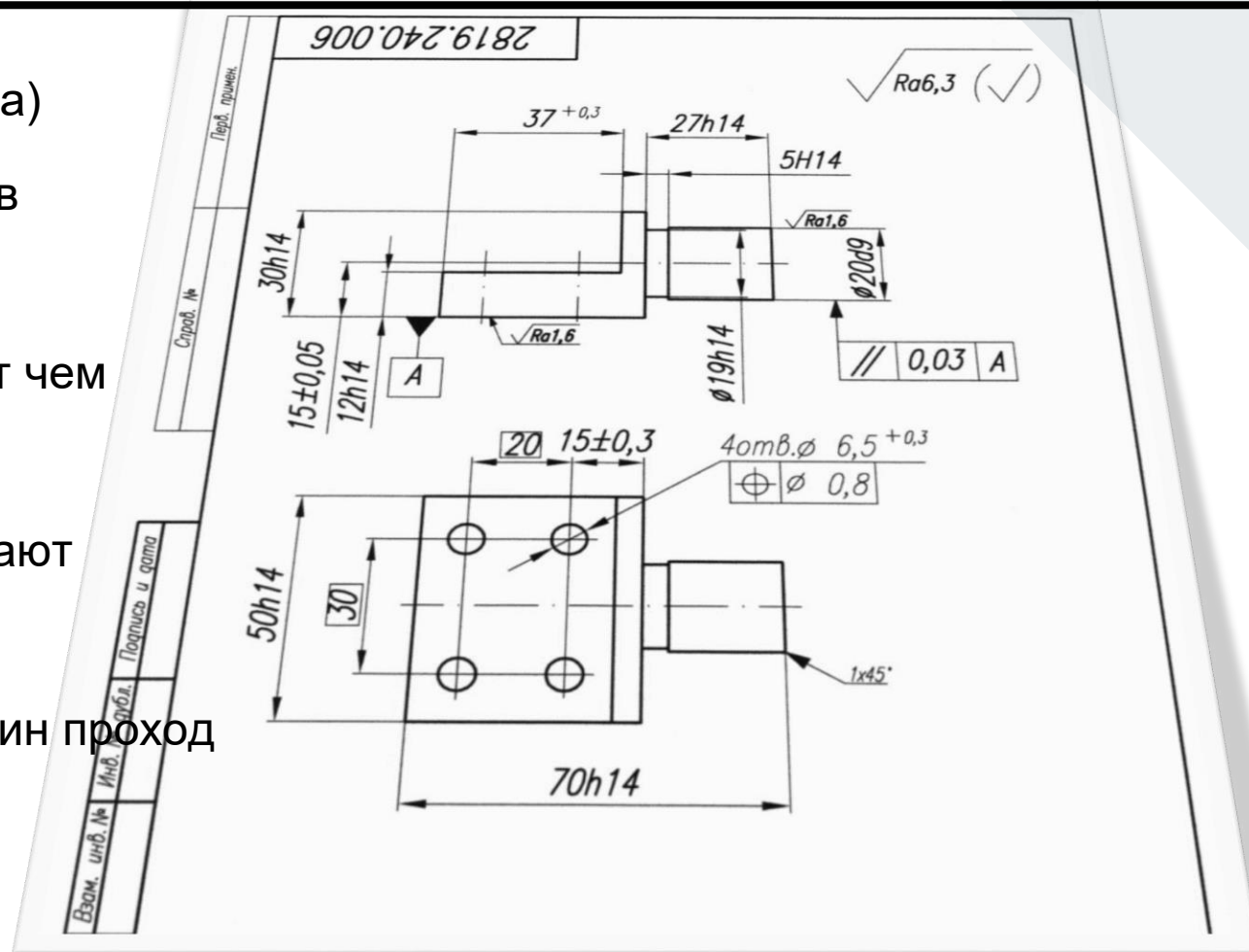
4x4x4

val_loss: 0.5399

val_accuracy: 0.7957

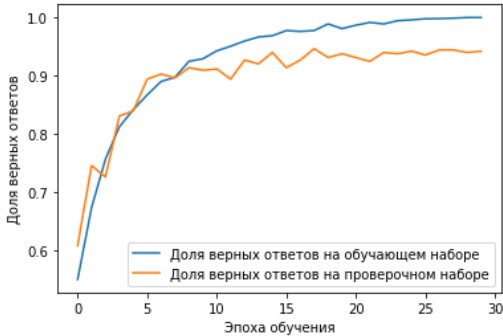
Особенности работы с изображением чертежа

- Большой размер исходного изображения(тензора)
- Высокое сжатие (64 раза и более)изображения в процессе обработки
- Слой MaxPooling2D показывает лучше результат чем параметр strides
- Слои AveragePooling2D, SeparableConv2D не дают эффекта
- Применение сжатие более чем в два раза за один проход не снижает качество работы сети



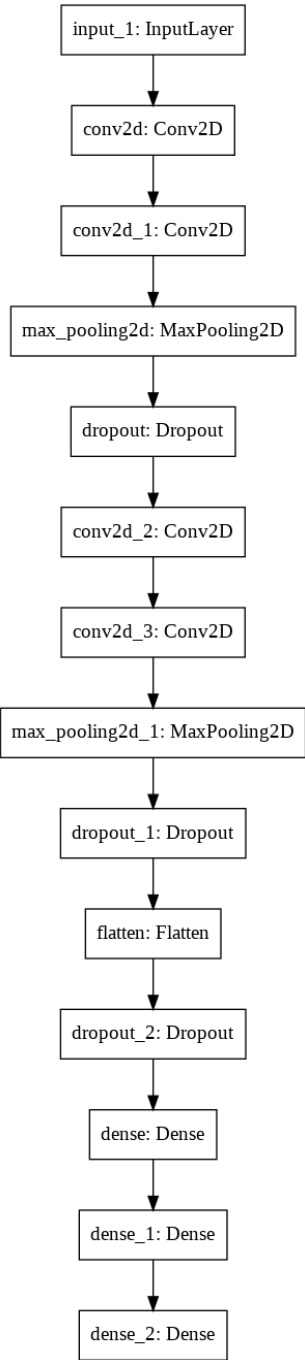
Последовательная сверточная нейронная сеть model a

conv2d	(Conv2D)
conv2d	(Conv2D)
max_pooling2d	(MaxPooling2D)
Dropout	(Dropout)

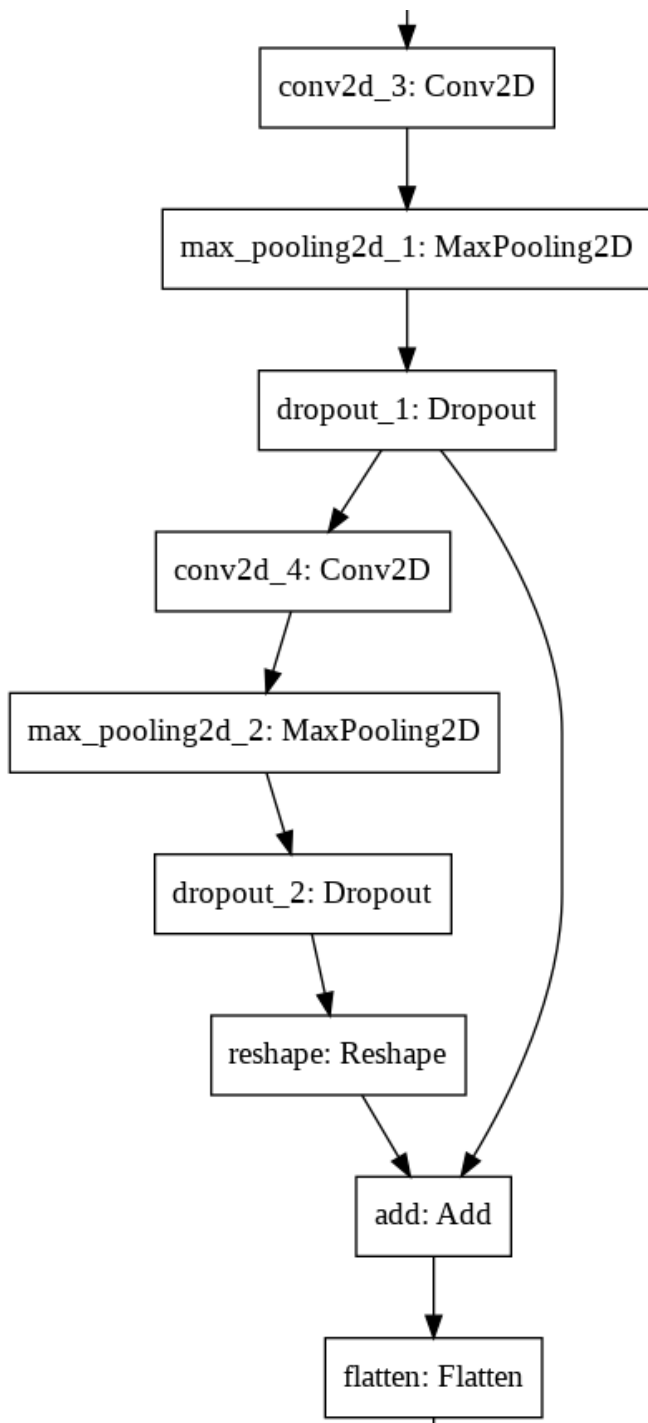


val_loss: 0.1863 - val_accuracy: 0.9457

Входной тензор	512x384x1
Выходной тензор	8x6x128
Степень сжатия	64
Количество слоев сжатия	8x8



Layer (type)	Output Shape
input_1 (InputLayer)	[(None, 512, 384, 1)]
conv2d (Conv2D)	(None, 512, 384, 8)
conv2d_1 (Conv2D)	(None, 512, 384, 32)
max_pooling2d (MaxPooling2D)	(None, 64, 48, 32)
dropout (Dropout)	(None, 64, 48, 32)
conv2d_2 (Conv2D)	(None, 64, 48, 64)
conv2d_3 (Conv2D)	(None, 64, 48, 128)
max_pooling2d_1 (MaxPooling2D)	(None, 8, 6, 128)
dropout_1 (Dropout)	(None, 8, 6, 128)
flatten (Flatten)	(None, 6144)
dropout_2 (Dropout)	(None, 6144)
dense (Dense)	(None, 512)
dense_1 (Dense)	(None, 256)
dense_2 (Dense)	(None, 3)



Модель сети с остаточными связями model b

Layer (type)	Output Shape	Param #
input_1 (InputLayer)	[(None, 512, 384, 1)]	0
conv2d (Conv2D)	(None, 512, 384, 8)	80
conv2d_1 (Conv2D)	(None, 512, 384, 32)	2336
max_pooling2d (MaxPooling2D)	(None, 64, 48, 32)	0
dropout (Dropout)	(None, 64, 48, 32)	0
conv2d_2 (Conv2D)	(None, 64, 48, 64)	18496
conv2d_3 (Conv2D)	(None, 64, 48, 128)	73856
max_pooling2d_1 (MaxPooling2D)	(None, 8, 6, 128)	0
dropout_1 (Dropout)	(None, 8, 6, 128)	0
conv2d_4 (Conv2D)	(None, 8, 6, 512)	590336
max_pooling2d_2 (MaxPooling2D)	(None, 4, 3, 512)	0
dropout_2 (Dropout)	(None, 4, 3, 512)	0
reshape (Reshape)	(None, 8, 6, 128)	0
add (Add)	(None, 8, 6, 128)	0
flatten (Flatten)	(None, 6144)	0
dropout_3 (Dropout)	(None, 6144)	0
dense (Dense)	(None, 512)	3146240
dense_1 (Dense)	(None, 256)	131328
dense_2 (Dense)	(None, 3)	771

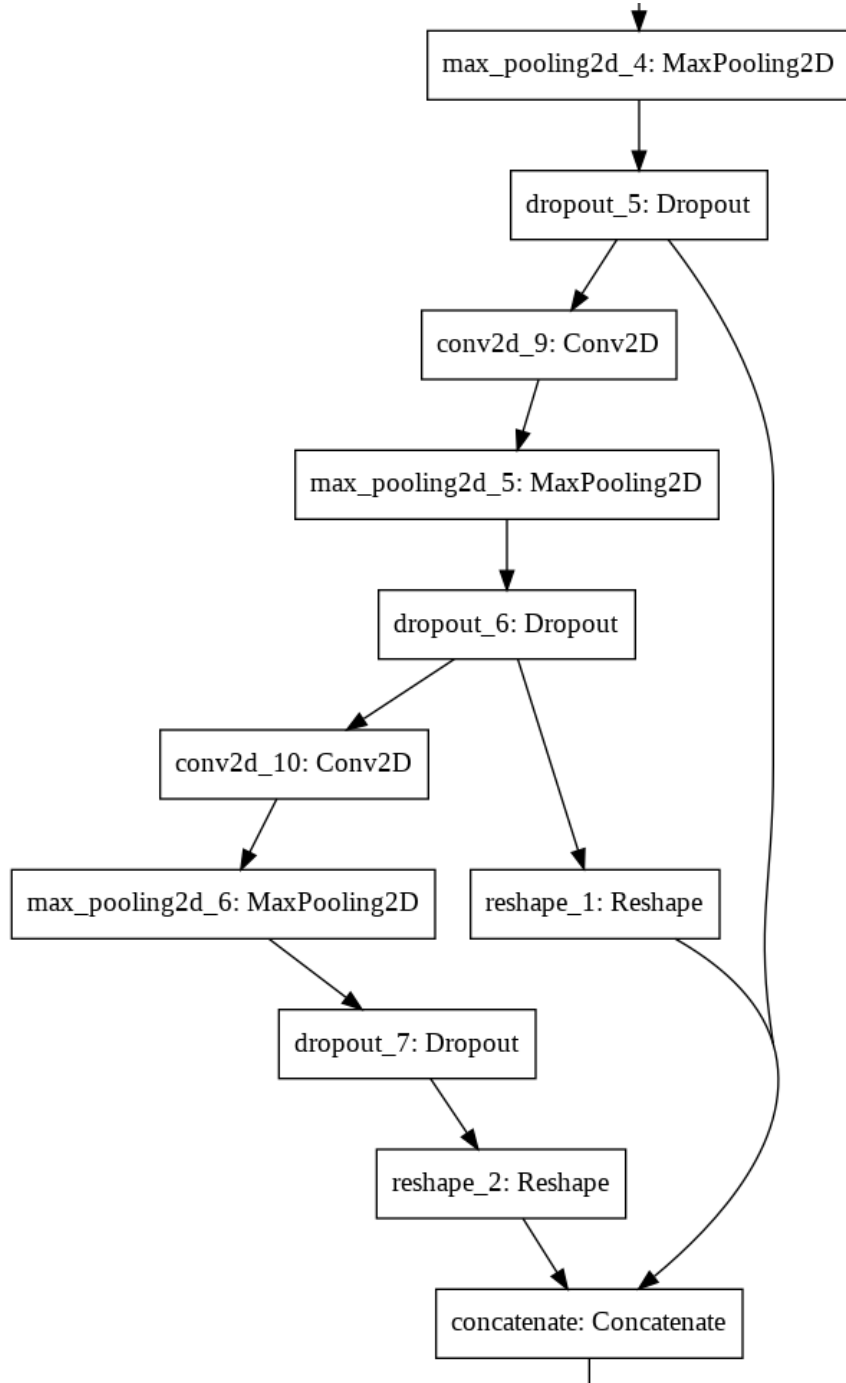
Объединяем два слоя методом add

$X1 = (\text{None}, 8, 6, 128)$

$X2 = (\text{None}, 4, 3, 512)$

$x = \text{add}([x1, x2])$

$x2 = \text{Reshape}((8, 6, 128), \text{input_shape} = (4, 3, 512))(x2)$



Модель сети с остаточными связями model c

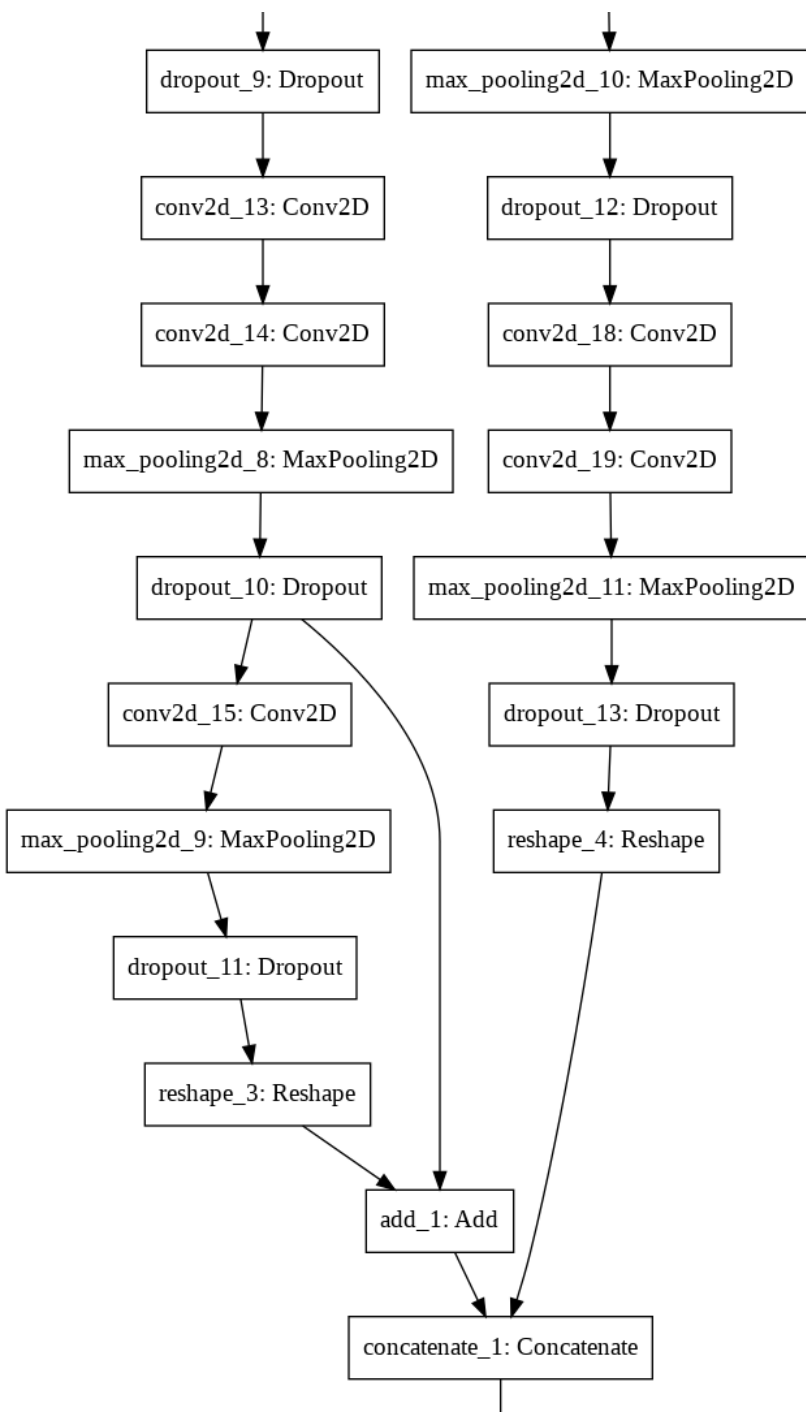
Объединяем три слоя методом concatenate

X1 = (None, 8, 6, 128)
X2 = (None, 4, 3, 512)
X3 = (None, 2, 1, 1536)

x = concatenate([x1, x2, x3])

max_pooling2d_4	(MaxPooling2D)	(None, 8, 6, 64)
dropout_5	(Dropout)	(None, 8, 6, 64)
conv2d_9	(Conv2D)	(None, 8, 6, 256)
max_pooling2d_5	(MaxPooling2D)	(None, 4, 3, 256)
dropout_6	(Dropout)	(None, 4, 3, 256)
conv2d_10	(Conv2D)	(None, 4, 3, 1536)
max_pooling2d_6	(MaxPooling2D)	(None, 2, 1, 1536)
dropout_7	(Dropout)	(None, 2, 1, 1536)
reshape_1	(Reshape)	(None, 8, 6, 64)
reshape_2	(Reshape)	(None, 8, 6, 64)
concatenate	(Concatenate)	(None, 8, 6, 192)

x2 = Reshape((8, 6, 64), input_shape = (4, 3, 256))(x2)
x3 = Reshape((8, 6, 64), input_shape = (2, 1, 1536))(x3)



Модель сети с несколькими параллельными ветвями model d

Объединяем model b с дополнительной ветвью методом concatenate

X1 = (None, 8, 6, 64)
X2 = (None, 4, 3, 256)

X3 = (None, 2, 1, 576)

x = add([x1, x2])

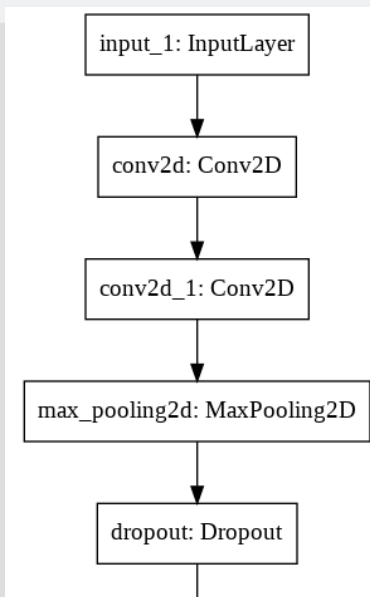
x = concatenate([x, x3])

conv2d_19 (Conv2D)	(None, 32, 24, 576)
dropout_11 (Dropout)	(None, 4, 3, 512)
max_pooling2d_11 (MaxPooling2D)	(None, 2, 1, 576)
reshape_3 (Reshape)	(None, 8, 6, 128)
dropout_13 (Dropout)	(None, 2, 1, 576)
add_1 (Add)	(None, 8, 6, 128)
reshape_4 (Reshape)	(None, 8, 6, 24)
concatenate_1 (Concatenate)	(None, 8, 6, 152)
flatten_2 (Flatten)	(None, 7296)

x2 = Reshape((8, 6, 64), input_shape = (4, 3, 256))(x2)
x3 = Reshape((8, 6, 24), input_shape = (2, 1, 576))(x3)

Ансамблирование моделей как способ решения

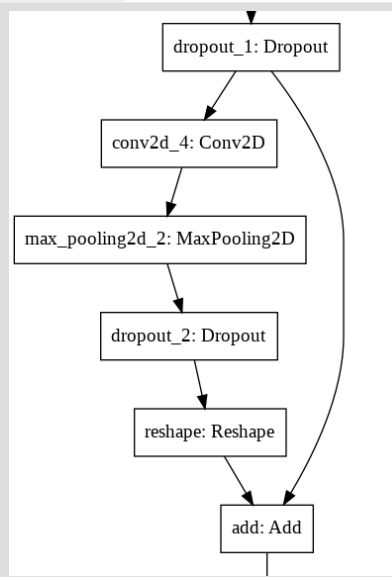
model a



val_loss: 0.1863

val_accuracy: 0.9456

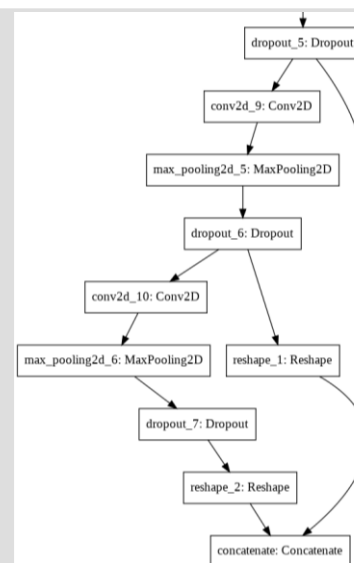
model b



val_loss: 0.1417

val_accuracy: 0.9543

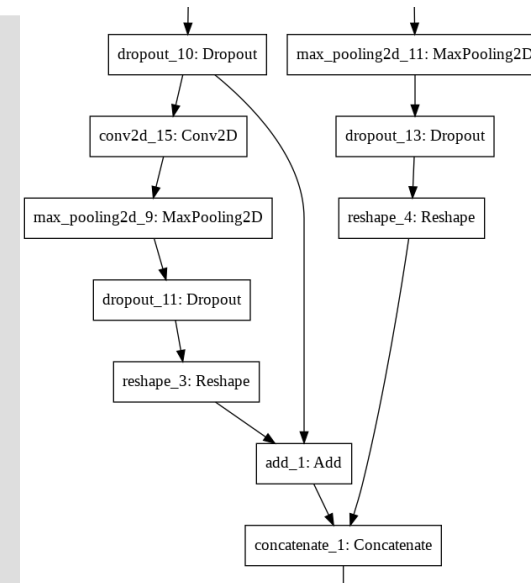
model c



val_loss: 0.1626

val_accuracy: 0.9478

model d



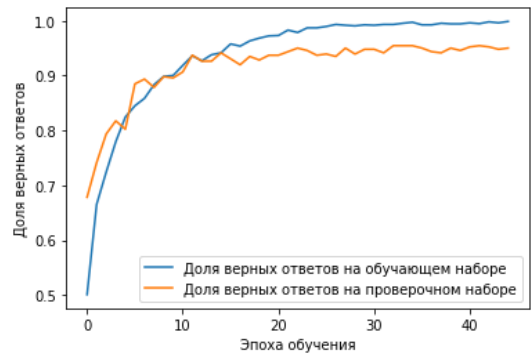
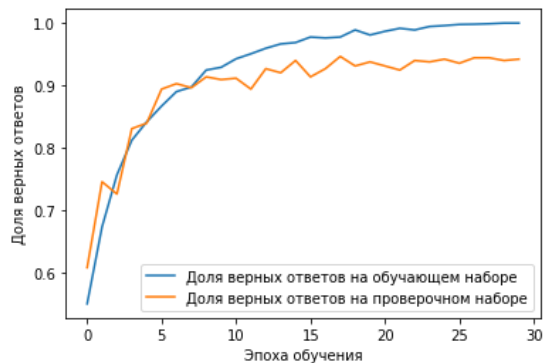
val_loss: 0.1298

val_accuracy: 0.9543

$$\text{final_preds} = 0.15 * \text{preds_a} + 0.2 * \text{preds_b} + 0.15 * \text{preds_c} + 0.5 * \text{preds_d}$$

Процент распознавания

	Распознано верно %			Средний %
	class1	class2	class3	
model a	95,5	93,6	93,9	94,3
model b	95,0	91,0	97,8	94,6
model c	95,5	93,6	94,4	94,5
model d	96,5	91,0	96,1	94,5
final_preds	98,0	92,3	96,7	95,7



Вывод

Сборная модель дает
лучший результат чем
каждая по отдельности

