Gesture Recognition Write up

<table>
<tr><th>Exp. No.</th><th>Model</th><th>Result</th><th>Decision + Explanation</th></tr>
<tr><td>1</td><td>Conv3D</td><td><table><tr><td>Train acc.</td><td>0.18</td></tr><tr><td>Val. acc.</td><td>0.21</td></tr><tr><td>Time taken</td><td>45 mins</td></tr><tr><td>Conclusion</td><td>Model not training properly</td></tr></table></td><td>Resized images were stored in an image variable and not passed to batch data variable, hence the accuracy was very low.
Image indices: even nos. excluding 30
Image size: 160,160
batch size: 55
epoch: 25</td></tr>
<tr><td>2</td><td>ResNet152</td><td><table><tr><td>Train acc.</td><td>0.21</td></tr><tr><td>Val. acc.</td><td>0.23</td></tr><tr><td>Time taken</td><td>56 mins</td></tr><tr><td>Conclusion</td><td>Model not training properly</td></tr></table></td><td>Tried with different network. But same issue of the resized images stored in an image variable and not passed to batch data variable, hence the accuracy was very low here as well.
Image indices: even nos. excluding 30
Image size: 160,160
batch size: 55
epoch: 25</td></tr>
<tr><td>3</td><td>ResNet152</td><td><table><tr><td>Train acc.</td><td>0.98</td></tr><tr><td>Val. acc.</td><td>0.71</td></tr><tr><td>Time taken</td><td>30 mins</td></tr><tr><td>Conclusion</td><td>Overfitting</td></tr></table></td><td>Reduced the batch size, from 55 to 15, resized and normalized image correctly. However, we could notice model was overfitting.
Image indices: even nos. excluding 30
Image size: 160,160
batch size: 15
epoch: 25</td></tr>
<tr><td>4</td><td>Conv3D</td><td><table><tr><td>Train acc.</td><td>0.94</td></tr><tr><td>Val. acc.</td><td>0.80</td></tr><tr><td>Time taken</td><td>20 mins</td></tr><tr><td>Conclusion</td><td>Overfitting</td></tr></table></td><td>Tried with conv3D and same parameters but had the same issue of overfitting
Image indices: even nos. excluding 30
Image size: 160,160
batch size: 15
epoch: 25</td></tr>
</table>

<table>
<tr><th>Exp. No.</th><th>Model</th><th>Result</th><th>Decision + Explanation</th></tr>
<tr><td>5</td><td>Conv3D</td><td><table><tr><td>Train acc.</td><td>0.95</td></tr><tr><td>Val. acc.</td><td>0.19</td></tr><tr><td>Time taken</td><td>7 mins</td></tr><tr><td>Conclusion</td><td>Overfitting and model stopped training because of early stopping.</td></tr></table></td><td>Added Early Stopping to reduce the overfitting. Reduced the image indices being used and also increased the batch size.
Image indices: multiples of 3 including 0 and excluding 30.
Image size: 160,160
batch size: 25
epoch: 25</td></tr>
<tr><td>6</td><td>ResNet152</td><td><table><tr><td>Train acc.</td><td>0.96</td></tr><tr><td>Val. acc.</td><td>0.67</td></tr><tr><td>Time taken</td><td>17 mins</td></tr><tr><td>Conclusion</td><td>Overfitting and model stopped training because of early stopping.</td></tr></table></td><td>Added Early Stopping on different model to reduce overfitting on it. Also reduced image size and batch size. Increased no. of epoch.
Image indices: multiples of 3 including 0 and excluding 30.
Image size: 120,120
batch size: 10
epoch: 40</td></tr>
<tr><td>7</td><td>ResNet152</td><td><table><tr><td>Train acc.</td><td>0.98</td></tr><tr><td>Val. acc.</td><td>0.76</td></tr><tr><td>Time taken</td><td>2 hrs 19 mins</td></tr><tr><td>Conclusion</td><td>Not optimal at all and overfitting is still there</td></tr></table></td><td>Increased image indices and image size in order to increase data, increased epoch as well.
Image indices: all images
Image size: 160,160
batch size: 10
epoch: 50</td></tr>
<tr><td>8</td><td>Conv3D</td><td><table><tr><td>Train acc.</td><td>0.75</td></tr><tr><td>Val. acc.</td><td>0.82</td></tr><tr><td>Time taken</td><td>12 mins</td></tr><tr><td>Conclusion</td><td>Underfitting</td></tr></table></td><td>Added extra Dropout layer, decreased image indices and reduced no. of epochs to reduce overfitting. But this led to underfitting.
Image indices: multiples of 3 including 0 and excluding 30.
Image size: 160,160
batch size: 10
epoch: 40

From this exp. onwards we decided to move ahead with Conv3D only because it was way faster.</td></tr>
</table>

<table>
<tr><td></td><td></td><td></td><td></td></tr>
<tr><td>Exp. No.</td><td>Model</td><td>Result</td><td>Decision + Explanation</td></tr>
<tr><td>9</td><td>Conv3D</td><td><table><tr><td>Train acc.</td><td>0.89</td></tr><tr><td>Val. acc.</td><td>0.83</td></tr><tr><td>Time taken</td><td>15 mins</td></tr><tr><td>Conclusion</td><td>Good fit, but can improve</td></tr></table></td><td>Removed the extra Dropout layer and kept other parameters same.
Image indices: multiples of 3 including 0 and excluding 30.
Image size: 160,160
batch size: 10
epoch: 40

This was a good fit, but we thought we could improve so did one last final step.</td></tr>
<tr><td>10</td><td>Conv3D</td><td><table><tr><td>Train acc.</td><td>0.94</td></tr><tr><td>Val. acc.</td><td>0.90</td></tr><tr><td>Time taken</td><td>34 mins</td></tr><tr><td>Conclusion</td><td>Good fit</td></tr></table></td><td>Augmented(cropped) images and kept the parameters same.
Image indices: multiples of 3 including 0 and excluding 30.
Image size: 160,160
batch size: 10
epoch: 40

This is the final model that we selected for submission.</td></tr>
</table>