

```

1: //////////////////////////////////////
2: Visual Cifar10 Test
3:
4: maXbox Starter 89_2 - Train with a pretrained model. - Max Kleiner
5:
6: "space your code" "pace your code"
7:   - mX4
8:   https://my6.code.blog/2021/09/08/improver-4/
9:   https://github.com/snatch59/load-cifar-10
10:
11: This tutor explains training the so called CIFAR-10 Image Classifier with loading a pretrained
    model.
12: This command line tool runs the CAI Network with CIFAR10 files. Utility to load cifar-10 image
    data into training and test data sets. Download the cifar-10 python version dataset from here,
    and extract the cifar-10-batches-1..5 folder into the same directory as the script.
13:
14: The code contains example usage, and runs under Python 3 and maXbox4 only. Note that the
    load_cifar_10_data() function has the option to load the images as negatives using
    negatives=True. We start with a CIFAR-10 SELU Classification Example Visual Learn (comments in
    code):
15:
16:
17: procedure TFormVisualLearningLearn(Sender: TObject);
18:   var
19:     NN: TNNNet;
20:     I: integer;
21:     ImgVolumes: TNNNetVolumeList;
22:     Volume: TNNNetVolume;
23:     pOutput, vOutput, vDisplay: TNNNetVolume;
24:     hit, miss: integer;
25:     NumClasses: integer;
26:     ErrorSum, LastError: TNeuralFloat;
27:     startTime, totalTimeSeconds: double;
28:     aImage: array of TImage;
29:     NeuronCount: integer;
30:     ImgIdx: integer;
31:     MaxW, MinW: TNeuralFloat;
32:     fileName: string;
33:     firstNeuronalLayer: integer;
34:     OpenFileDialogNN: TOpenDialog;
35:     FormVisualLearning: TForm;
36:     FRunning: boolean;
37:     ButTest: TButton;
38:     EdTestBinFile: TEdit;
39:     ImgSample: TImage;
40:     LabClassRate: TLabel;
41:     LabTestFile: TLabel;
42:     //OpenDialogNN: TOpenDialog;
43:
44:
45: begin
46:   writeln('Creating Neural Network...');
47:   ImgVolumes:= TNNNetVolumeList.Create(true);
48:   NumClasses:= 10;
49:   frunning:= true;
50:
51:   fileName:= 'cnnmyvisualfilename_pretrain.nn' ;//OpenDialogNN.FileName;
52:
53:   //-----
54:   // creates required volumes to store cifar images
55:   for I:= 0 to 9999 do begin
56:     Volume:= TNNNetVolume.Create();
57:     ImgVolumes.Add(Volume);
58:   end;
59:   //-----
60:
61:   NN:= TNNNet.Create();
62:   //Nn.verbose:= true;
63:
64:   FormVisualLearning:= TForm.create(self);
65:   FormVisualLearning.setbounds(100,100,400,400)
66:   FormVisualLearning.show;
67:   ImgSample:= TImage.create(self);
68:   imgsampl.align:= alclient;
69:   try

```

```

70:  writeln('Loading neural network from file: '+fileName);
71:  NN.LoadFromFile('C:\maXbox\EKON_BASTA\EKON24\cifar-10-batches-
bin\ImageClassifierSELU_Tutor89.nn');
72:  NN.EnabledDropouts(false);
73:  firstNeuronalLayer:= NN.GetFirstNeuronalLayerIdx(1);
74:
75:  pOutput := TNNNetVolume.Create0(NumClasses,1,1,0);
76:  vOutput := TNNNetVolume.Create0(NumClasses,1,1,0);
77:  vDisplay:= TNNNetVolume.Create0(NumClasses,1,1,0);
78:
79:  SetLength(aImage, NN.Layers[firstNeuronalLayer].Neurons.Count);
80:
81:  for NeuronCount:= 0 to NN.Layers[firstNeuronalLayer].Neurons.Count - 1 do begin
82:    aImage[NeuronCount]:= TImage.Create(FormVisualLearning);
83:    aImage[NeuronCount].Parent:= FormVisualLearning;
84:    aImage[NeuronCount].Width :=
NN.Layers[firstNeuronalLayer].Neurons[NeuronCount].Weights.SizeX;
85:    aImage[NeuronCount].Height:=
NN.Layers[firstNeuronalLayer].Neurons[NeuronCount].Weights.SizeY;
86:    aImage[NeuronCount].Top := (NeuronCount div 12) * 36 + 120;
87:    aImage[NeuronCount].Left := (NeuronCount mod 12) * 36 + 32;
88:    aImage[NeuronCount].Stretch:=true;
89:  end;
90:
91:  NN.DebugWeights();
92:  WriteLn('Neural network has: ');
93:  WriteLn(' Layers: '+itoa( NN.CountLayers() ));
94:  WriteLn(' Neurons:'+itoa( NN.CountNeurons() ));
95:  WriteLn(' Weights:'+itoa( NN.CountWeights() ));
96:  WriteLn('N Computing...');
97:
98:  begin
99:    hit := 0;
100:    miss := 0;
101:    ErrorSum := 0;
102:    LastError:= 0;
103:    startTime:= Now();
104:    //loadCifar10Dataset7(ImgVolumes,
105:    // 'C:\maXbox\EKON_BASTA\EKON24\cifar-10-batches-bin\data_batch_1.bin',0,csEncodeRGB);
106:    //1 is the index of data_batch_1.bin
107:    loadCifar10Dataset6(ImgVolumes, 1, 0, csEncodeRGB);
108:    for I:= 0 to ImgVolumes.Count - 1 do begin
109:      if not(FRunning) then Break;
110:      ImgIdx := Random(ImgVolumes.Count);
111:      //-- CAREFUL
112:      NN.Compute65(ImgVolumes[ImgIdx],0);
113:      NN.GetOutput(pOutput);
114:      //showmessage('test this1 running')
115:
116:      vOutput.SetClassForReLU( ImgVolumes[ImgIdx].Tag ); // ReLU - no softmax
117:      ErrorSum:= errorsum+ vOutput.SumDiff(pOutput);
118:
119:      if I mod 1000 = 0 then begin
120:        vDisplay.Copy38(ImgVolumes[ImgIdx]);
121:        vDisplay.Mul26(64);
122:        vDisplay.Add13(128);
123:
124:        LoadVolumeIntoTImage(vDisplay, ImgSample, csEncodeRGB);
125:        ImgSample.Width := 64;
126:        ImgSample.Height:= 64;
127:
128:        for NeuronCount:= 0 to NN.Layers[firstNeuronalLayer].Neurons.Count - 1 do begin
129:          MaxW:= NN.Layers[firstNeuronalLayer].Neurons[NeuronCount].Weights.GetMax();
130:          MinW:= NN.Layers[firstNeuronalLayer].Neurons[NeuronCount].Weights.GetMin();
131:          vDisplay.Copy38(NN.Layers[firstNeuronalLayer].Neurons[NeuronCount].Weights);
132:          vDisplay.Mul26(256/(MaxW-MinW));
133:          vDisplay.Add13(128);
134:          LoadVolumeIntoTImage(vDisplay, aImage[NeuronCount], csEncodeRGB);
135:          aImage[NeuronCount].Width := 32;
136:          aImage[NeuronCount].Height:= 32;
137:          writeln('neuron count: '+itoa(neuroncount));
138:        end;
139:        Application.ProcessMessages();
140:      end;
141:

```

```

142:     Application.ProcessMessages();
143:
144:     if pOutput.GetClass() = ImgVolumes[ImgIdx].Tag then begin
145:         Inc(Hit);
146:     end else begin
147:         Inc(Miss);
148:     end;
149:
150:     if (Hit>0) and (I>0) and ((I+1) mod 1000 = 0) then begin
151:         totalTimeSeconds:= (Now() - startTime) * 24 * 60 * 60;
152:         WriteLn (
153:             '+'
154:             itoa(I+1)+ ' Accuracy:'+flots( Hit/(Hit+Miss))+
155:             ' Error:'+flots(ErrorSum-LastError)+
156:             ' Time:'+ flots(totalTimeSeconds)+'s'+
157:             ' Forward:'+flots (NN.ForwardTime * 24 * 60 * 60)+'s'+
158:             ' Backward:'+flots(NN.BackwardTime * 24 * 60 * 60)+'s' // *)
159:         );
160:         NN.ClearTime();
161:
162:         //LabClassRate.Caption:= IntToStr( (Hit*100) div (Hit+Miss) )+'%';
163:         writeln(IntToStr( (Hit*100) div (Hit+Miss) )+'%');
164:         startTime:= Now();
165:         LastError:= ErrorSum;
166:         Application.ProcessMessages;
167:     end;
168: end;
169:
170:     NN.DebugWeights();
171: end;
172:
173: for NeuronCount := Low(aImage) to High(aImage) do begin
174:     aImage[NeuronCount].Free;
175: end;
176:
177: //LabClassRate.Caption := '0%';
178: //writeln(LabClassRate.Caption := '0%');
179: finally
180:     vDisplay.Free;
181:     NN.Free;
182:     vOutput.Free;
183:     pOutput.Free;
184:     ImgVolumes.Free;
185:     ImgSample.Free;
186:     FormVisualLearning.Free;
187:     writeln('free neuralnet and volumes');
188: end;
189: end;
190:
191: Script Ref: 1073__CAI_3_LearnerClassifier22_Tutor_89_2.txt
192:
193: https://sourceforge.
    net/p/cai/svncode/HEAD/tree/trunk/lazarus/experiments/visualCifar10test/uvisualcifar10test.pas
194: http://www.softwareschule.ch/examples/1073__CAI_3_LearnerClassifier22_Tutor_89_2.txt
195: https://entwickler-konferenz.de/blog/machine-learning-mit-cai/
196: https://www.freecodecamp.org/news/convolutional-neural-network-tutorial-for-beginners/

```