Face Recognition Using the Eigenface Algorithm

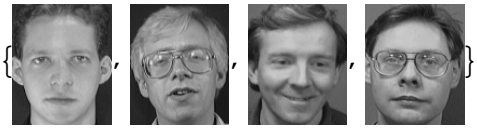
Initialization Code (optional)

Manipulate

```
Speak["Who is this?"]
```

In[1]:=

```
Manipulate[
 Module[{y, b, j, m, c, v, u, ω, h, l, m1, w, q, data, i},
  y = Table[0, {i, 4}];
  b = Table[0, {i, 10304}, {j, 4}];
```

y =  ;

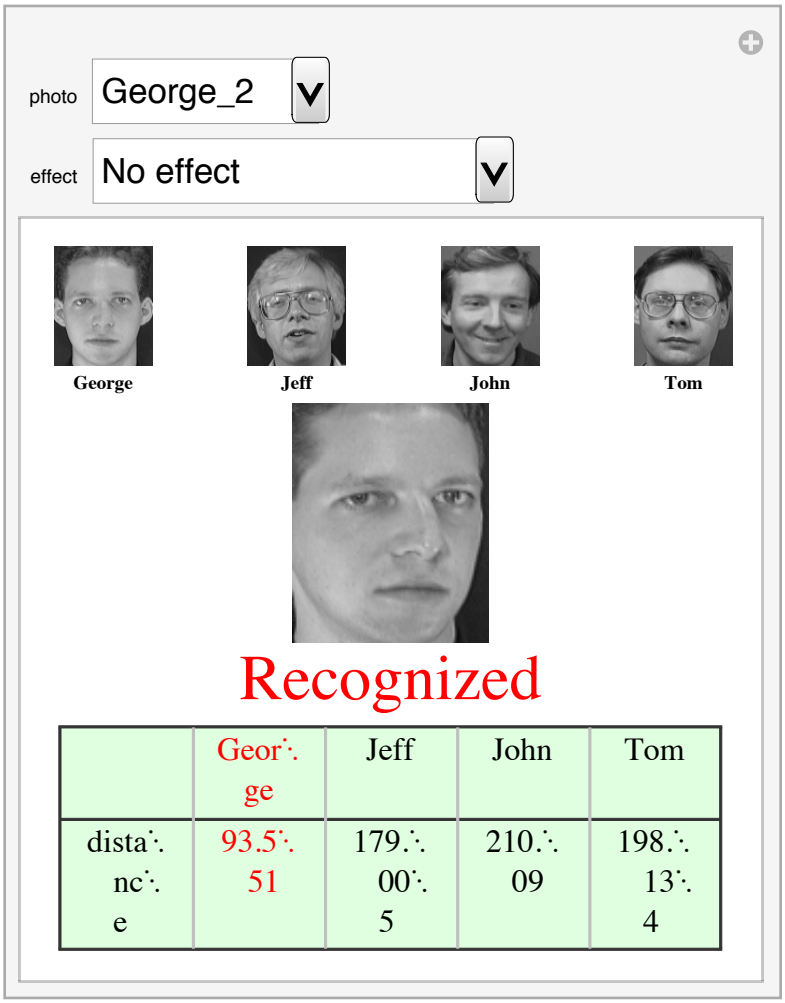
```
  For[j = 1, j ≤ 4, j++, b[[All, j]] = Flatten[ImageData[y[[j]]]]];
  m = Table[0, {i, 10304}];
  For[j = 1, j ≤ 4, j++, m += 1/4 * b[[All, j]]];
  For[j = 1, j <= 4, j++, b[[All, j]] = b[[All, j]] - m];
  c = Transpose[b].b;
  v = Eigenvectors[c];
  u = b.Transpose[v];
  ω = Transpose[u].b;
  h = Table[0, {i, 24}];
  l = Table[0, {i, 10304}, {j, 24}];
```

h =  ;

```
  h[[x]] = If[effect ≠ {}, ImageEffect[h[[x]], effect], h[[x]]];
  For[j = 1, j ≤ 24, j++, l[[All, j]] = Flatten[ImageData[h[[j]]]]];
  m1 = Table[0, {i, 10304}];
  For[j = 1, j ≤ 24, j++, m1 += 1/24 * l[[All, j]]];
  For[j = 1, j ≤ 24, j++, l[[All, j]] = l[[All, j]] - m1];
  w = Transpose[u].l;
  q = Table[0, {j, 4}];
  For[i = 1, i <= 4, i++, q[[i]] = Norm[w[[All, x]] - ω[[All, i]]] ];
  data = {{"distance", q[[1]], q[[2]], q[[3]], q[[4]]}};
  Column[{Row[{Column[{y[[1]], Text@Style["George", Bold]}, Center],
      "\t\t", Column[{y[[2]], Text@Style["Jeff", Bold]}, Center],
      "\t\t", Column[{y[[3]], Text@Style["John", Bold]}, Center],
      "\t\t", Column[{y[[4]], Text@Style["Tom", Bold]}, Center]}],
    Column[{h[[x]], Text[If[Min @@ q == q[[IntegerPart[(x - 1)/6] + 1]],
```

```
        Style["Recognized", Red, 30], Style["Failed", Red, 30]]]}, Center],
    Text@Grid[Prepend[data, {"", "George", "Jeff", "John", "Tom"}], Background → {None, LightGreen },
      Dividers → {{Darker[Gray, .6], {Lighter[Gray, .5]}, Darker[Gray, .6]},
        {Darker[Gray, .6], Darker[Gray, .6], {False}, Darker[Gray, .6]}},
      Alignment → {{{Center}, Center, Center, Center}}, ItemSize → {{5, 5, 5, 5, 5}}, Frame → Darker[Gray, .6],
      ItemStyle → {{15, If[Min @@ q == q[[1]], {{15, Red}, {15, Black}, {15, Black}, {15, Black}},
          If[Min @@ q == q[[2]], {{15, Black}, {15, Red}, {15, Black}, {15, Black}},
           If[Min @@ q == q[[3]], {{15, Black}, {15, Black}, {15, Red}, {15, Black}}, If[Min @@ q == q[[4]],
             {{15, Black}, {15, Black}, {15, Black}, {15, Red}}]]]]}, Spacings → {2.3, 1}]}, Center]],
 {{x, 1, "photo"}, {1 → "George_1", 2 → "George_2", 3 → "George_3", 4 → "George_4", 5 → "George_5",
   6 → "George_6", 7 → "Jeff_1", 8 → "Jeff_2", 9 → "Jeff_3", 10 → "Jeff_4", 11 → "Jeff_5", 12 → "Jeff_6",
   13 → "John_1", 14 → "John_2", 15 → "John_3", 16 → "John_4", 17 → "John_5", 18 → "John_6", 19 → "Tom_1",
   20 → "Tom_2", 21 → "Tom_3", 22 → "Tom_4", 23 → "Tom_5", 24 → "Tom_6"}, ControlType → PopupMenu},
 {{effect, {}, "effect"}, {{} → "No effect", {"Charcoal", 1} → "Charcoal,1",
   {"Charcoal", 2} → "Charcoal,2", {"OilPainting", 2} → "OilPainting,2",
   {"OilPainting", 4} → "OilPainting,4", {"OilPainting", 6} → "OilPainting,6",
   {"SaltPepperNoise", 0.1} → "SaltPepperNoise,0.1", {"SaltPepperNoise", 0.2} → "SaltPepperNoise,0.2",
   {"SaltPepperNoise", 0.5} → "SaltPepperNoise,0.5", {"SaltPepperNoise", 0.75} → "SaltPepperNoise,0.75",
   {"Solarization"} → "Solarization"}, ControlType → PopupMenu},
 TrackedSymbols :→ {x, effect}]
```

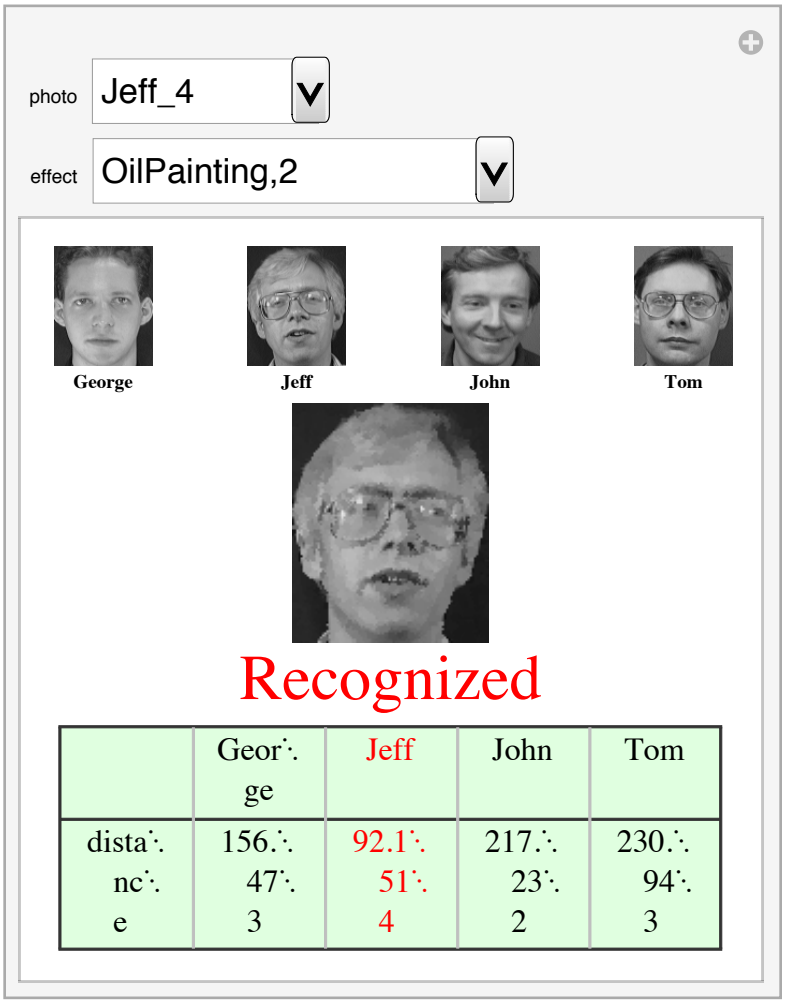
Out[1]=



Caption

There are many situations when we need to extract some information from a face database. With the assistance of the eigenface algorithm, we can identify a person from a photo not in the database.

Thumbnail



	George	Jeff	John	Tom
distance	156.47.3	92.151.4	217.23.2	230.94.3

Snapshots

photo John_3

effect No effect


George


Jeff


John


Tom

Recognized

	Geor⋱ge	Jeff	John	Tom
dista⋱nc⋱e	252.⋱99⋱6	253.⋱64⋱1	47.8⋱73⋱1	158.⋱43⋱4

photo John_3

effect Charcoal,2


George


Jeff


John


Tom

Failed

	Geor⋱ge	Jeff	John	Tom
dista⋱nc⋱e	102.⋱52⋱8	191.⋱99⋱6	213.⋱34⋱6	180.⋱69

photo John_3

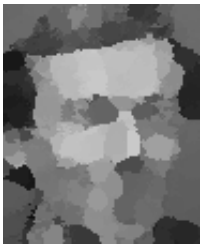
effect OilPainting,6


George


Jeff


John


Tom



Recognized

	George	Jeff	John	Tom
distance	248.003	256.728	50.927	154.653

photo John_3

effect SaltPepperNoise,0.2


George

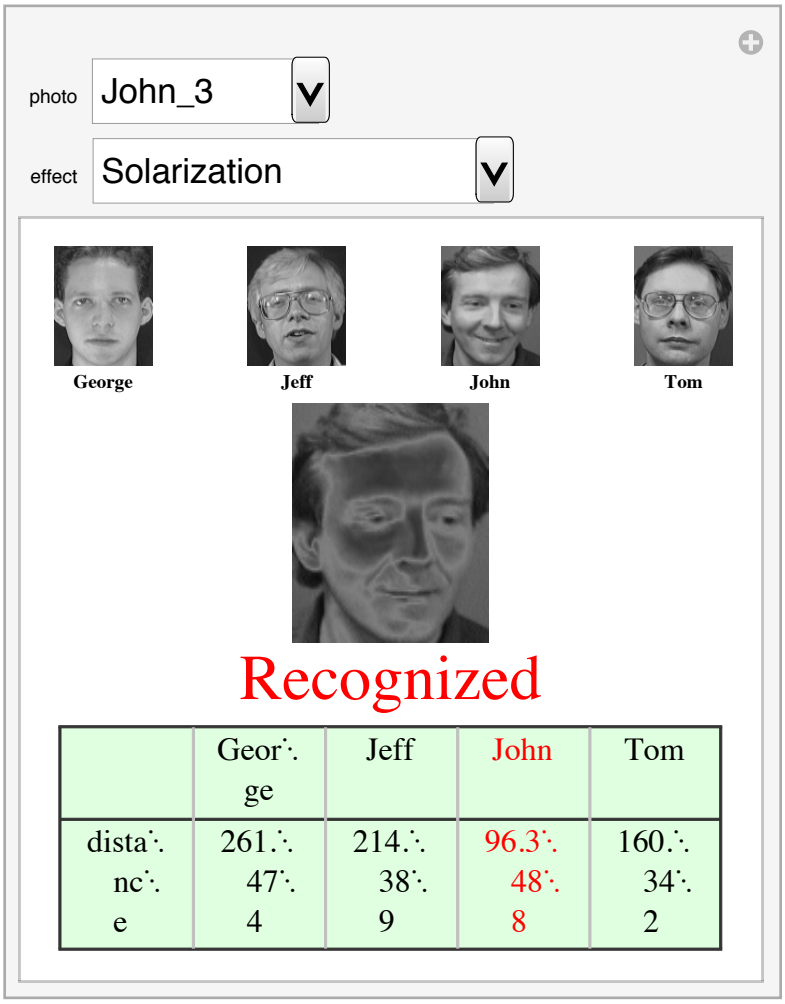

Jeff


John


Tom

Recognized

	George	Jeff	John	Tom
distance	240.44 6	237.18 1	66.052 6	151.98 8



Details (optional)

We have a face database of four people with photos of size 112×92 (10304 pixels). We represent each photo as a vector with 10304 elements to form a 10304×4 matrix A. From each element of A we subtract the mean value of the elements from the same row to get a matrix B. The four eigenvectors of the matrix $B^{T} \cdot B$ are calculated as are the coordinates of every "photo vector" in the basis formed by the eigenvectors.
Getting a set of 24 photos (six different photos of every person) we choose one of them, add some image effects if we want, and calculate its coordinates. It will correspond to one of the four photos we have with the minimum "distance" between them. So the given photo will be recognized or not.

Control Suggestions (optional)

- [x] Resize Images
- [ ] Rotate and Zoom in 3D
- [ ] Drag Locators
- [ ] Create and Delete Locators
- [ ] Slider Zoom

☐ Gamepad Controls

☐ Automatic Animation

☐ Bookmark Animation

Search Terms (optional)

face recognition
eigenface
image processing
image effects

Related Links (optional)

Authoring Information

Contributed by: Igor Mandric and Ion Andries