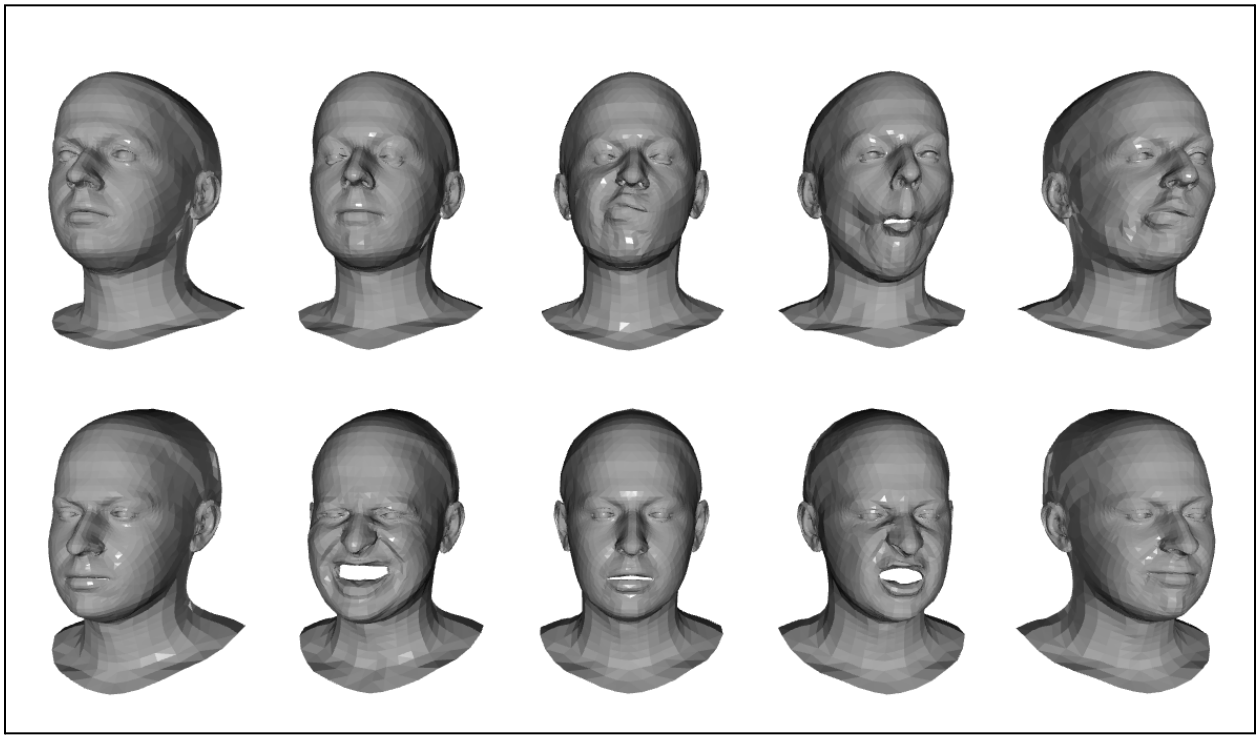
IF5171

Pembelajaran Mesin DSAI

Nama: Farrel Satya Putra Mahendra

NIM: 23522009

1. Visualisasikan wajah 3D *mesh* original minimal 10 wajah (10 poin).



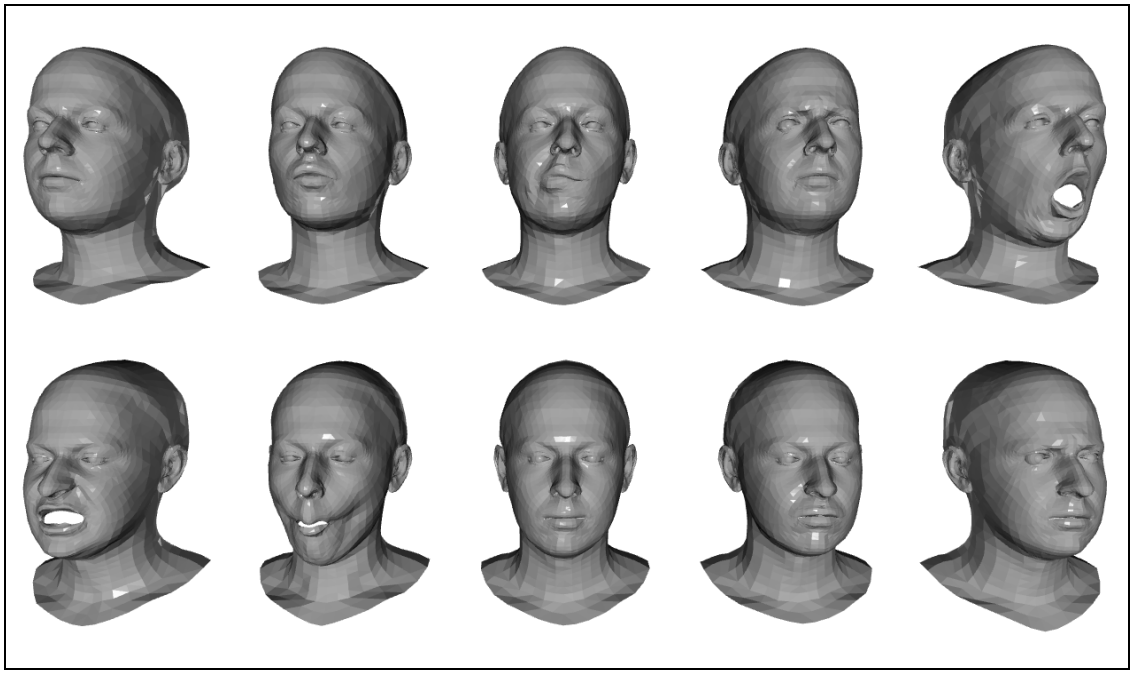
2. Bentuk / latih model wajah 3D secara *unsupervised* dengan menggunakan PCA. Pisahkan dataset asal menjadi 90% data *training* dan 10% data *test*. Ukurlah performa model dengan membandingkan wajah asli dan wajah rekonstruksi baik pada data training maupun data *test*, secara kuantitatif dengan *mean squared error*.

```
In [8]:  1 from sklearn.model_selection import train_test_split

In [9]:  1 data_train, data_test = train_test_split(dataset, test_size=0.1, random_state=42)

In [10]: 1 print('Total train:', len(data_train))
         2 print('Total test:', len(data_test))

Total train: 1064
Total test: 119
```

```
inversed_vertices_train_df = pd.DataFrame(inversed_vertices_train)
inversed_vertices_test_df = pd.DataFrame(inversed_vertices_test)

print("MSE Train:", mean_squared_error(vertices_train_df, inversed_vertices_train_df))
print("MSE Test:", mean_squared_error(vertices_test_df, inversed_vertices_test_df))
```

```
MSE Train: 8.418040921770731e-09
MSE Test: 6.375274903753317e-10
```

3. Visualisasikan hasil rekonstruksi wajah 3D *mesh* dari beberapa sampel pada data *test*, minimal 10 wajah (10 poin).

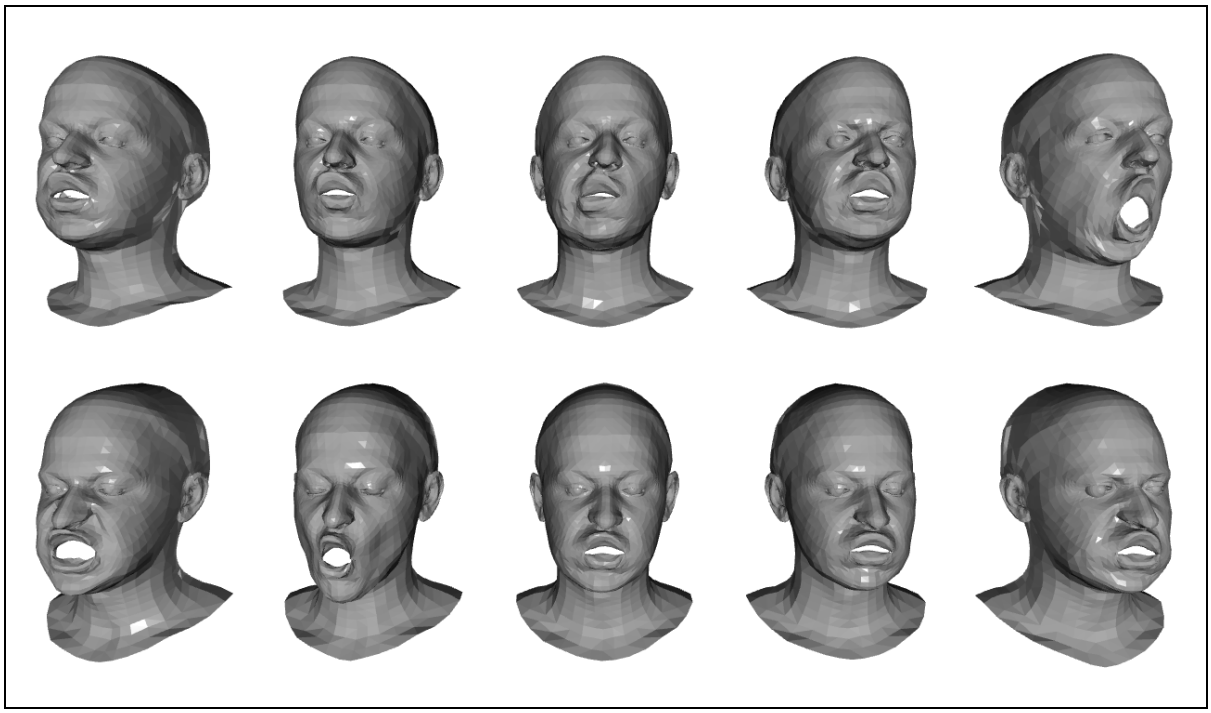


4. Visualisasikan dekodifikasi wajah 3D *mesh* dengan mengubah-ubah nilai salah satu kode atau elemen pada variabel *latent* z. Lalu dekodifikasikan 3D *mesh* terkait: x^=fdec(z) minimal 5 wajah (10 poin).

```
train_components_df[5] = 0.2
test_components_df[5] = 0.2
```

```
test_components_df.head(10)
```

	0	1	2	3	4	5	6	7	8	9	...	90	91	92	93	94
0	-0.304185	0.296476	0.116393	-0.020638	-0.026603	0.2	-0.022372	-0.011563	0.012371	0.002216	...	0.000191	-0.000799	0.000169	0.000373	-0.000430
1	-0.226014	-0.117398	-0.128399	0.132194	0.062564	0.2	0.014799	-0.008130	0.001470	0.006010	...	-0.000797	-0.001977	0.000814	-0.001539	-0.000409
2	-0.241554	0.069496	-0.069971	0.011131	-0.021923	0.2	-0.015185	-0.018618	-0.003265	-0.017363	...	0.002819	-0.001035	0.001388	-0.000291	-0.002368
3	-0.235915	-0.063871	0.161037	0.003271	-0.048055	0.2	-0.024539	-0.017103	0.000062	0.010350	...	-0.000607	0.000427	-0.000280	0.000421	0.000841
4	-0.211970	-0.045098	-0.086172	0.012568	-0.030065	0.2	-0.036575	0.013960	0.007604	-0.008844	...	0.000724	0.000312	-0.000608	0.001078	-0.001317
5	-0.209169	-0.097251	-0.049430	-0.019345	-0.017877	0.2	-0.023497	0.009405	0.014754	-0.015146	...	-0.000604	-0.000686	0.000143	0.000515	0.001217
6	-0.229410	-0.106802	-0.115802	0.172344	0.064978	0.2	0.014131	-0.008412	0.004113	0.006040	...	0.001351	0.002297	-0.000753	0.001944	0.000181
7	-0.316145	0.403390	-0.011342	-0.017930	0.016509	0.2	0.032439	0.000843	-0.034472	-0.001938	...	0.001587	-0.001823	-0.000324	0.000066	0.000834
8	-0.216147	-0.089987	-0.041672	-0.011741	-0.011171	0.2	-0.019350	0.012623	0.019304	0.001509	...	-0.000190	0.000268	-0.000798	0.000469	-0.000717
9	-0.214211	-0.066095	-0.056034	-0.005977	-0.060585	0.2	0.060187	0.001526	0.004428	0.009842	...	-0.001053	0.001629	-0.001073	-0.002188	0.000057



5. Visualisasikan 20 *principal components* dengan eigenvalues tertinggi (5 poin).

```
eigenvalues_train = pca_train.explained_variance_
eigenvalues_test = pca_test.explained_variance_

eigenvalues_train[:20]
```

```
array([2.67007474e+01, 1.67757558e+00, 3.37208588e-02, 2.02075024e-02,
       5.06421609e-03, 2.20436663e-03, 1.47642034e-03, 7.90686351e-04,
       5.03488131e-04, 4.69131645e-04, 3.11621271e-04, 2.43905381e-04,
       2.02006677e-04, 1.28602294e-04, 1.10036802e-04, 9.06904595e-05,
       7.45165379e-05, 6.41310780e-05, 5.86637603e-05, 4.89917827e-05])
```

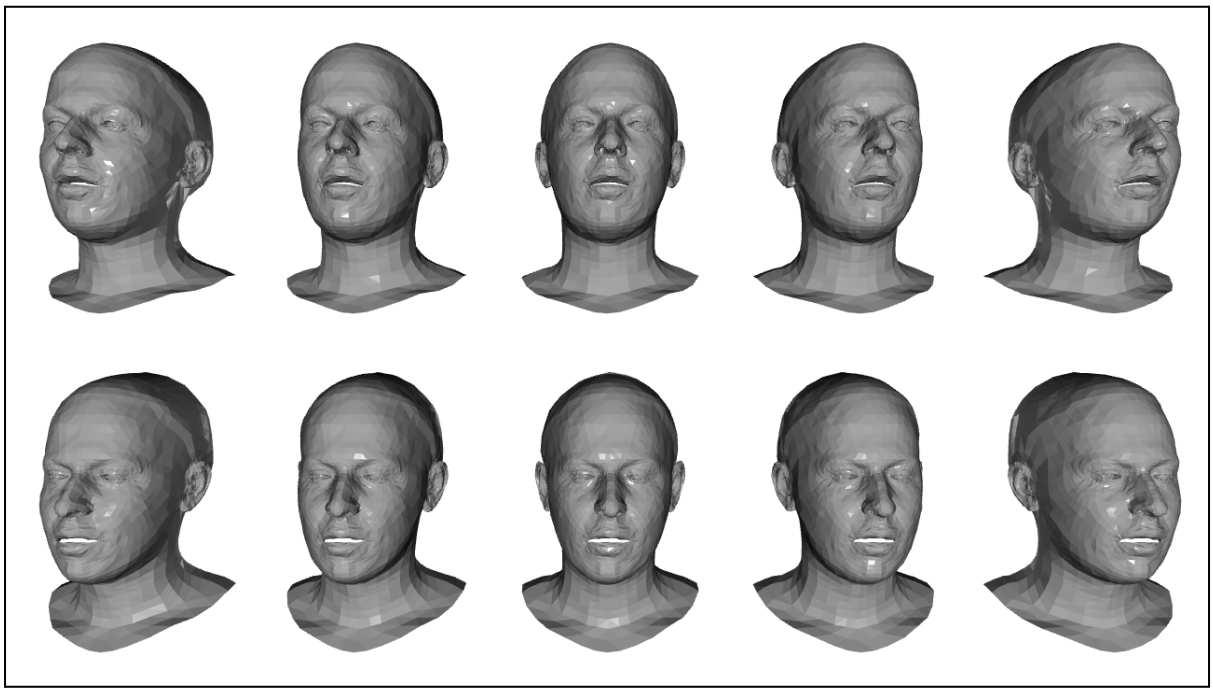
6. Ulangi langkah 2 - 4 dengan menggunakan Autoencoder (40 poin).
   a. Latih model wajah 3D secara unsupervised dengan *autoencoder*. Ukurlah performa model dengan membandingkan wajah asli dan wajah rekonstruksi baik pada data training maupun data *test*, secara kuantitatif dengan *mean squared error*.

```
decoded_train_df = decode_data(latent_train_df, autoencoder)
decoded_test_df = decode_data(latent_test_df, autoencoder)
```

```
print("MSE Train:", mean_squared_error(vertices_train_df, decoded_train_df))
print("MSE Test:", mean_squared_error(vertices_test_df, decoded_test_df))
```

```
MSE Train: 0.000152322758943767
MSE Test: 0.000433487933580046
```

   b. Visualisasikan hasil rekonstruksi wajah 3D *mesh* dari beberapa sampel pada data *test*.



c. Visualisasikan wajah 3D *mesh* dengan mengubah nilai salah satu kode atau elemen pada variabel *latent*.