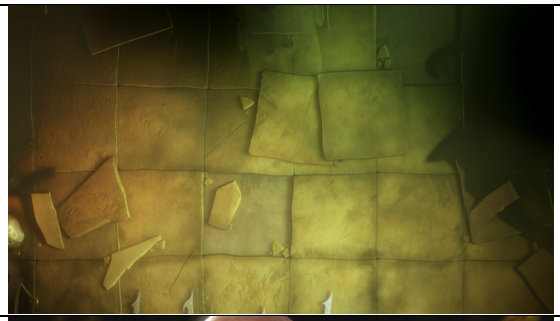
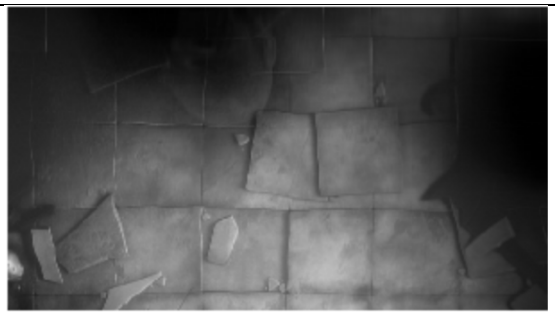
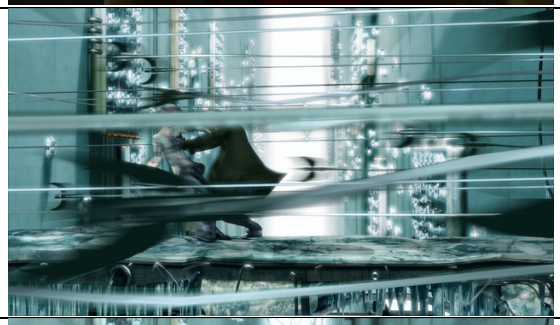
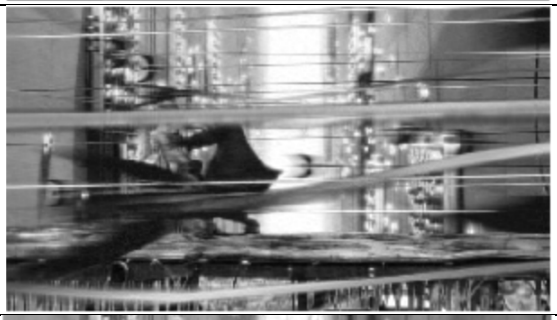
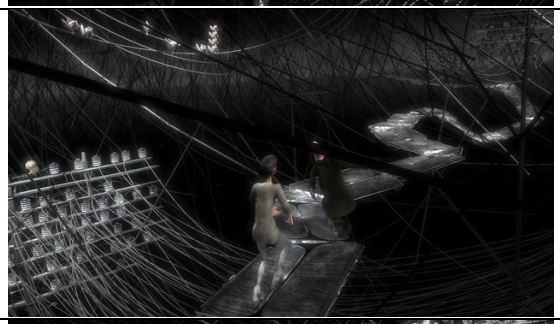
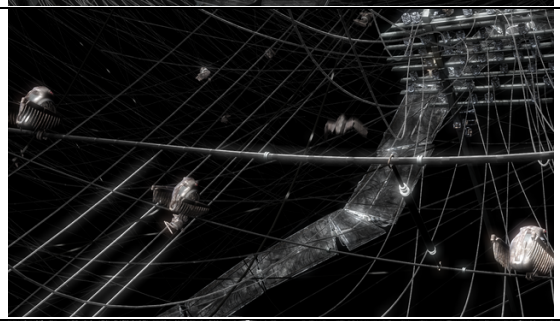
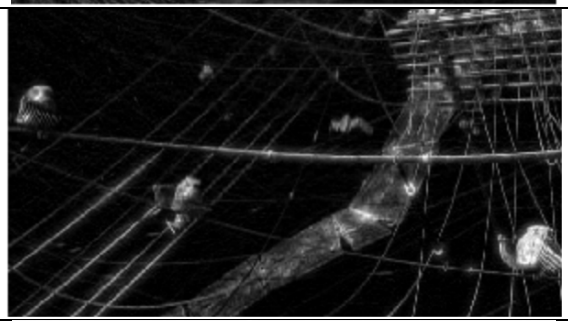
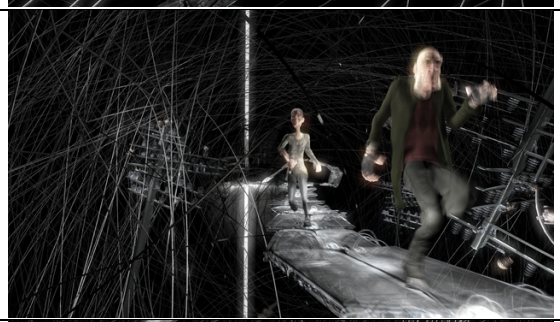
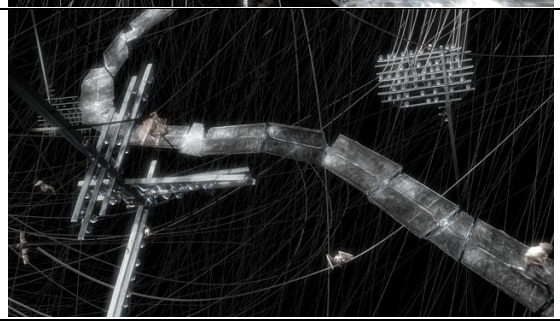
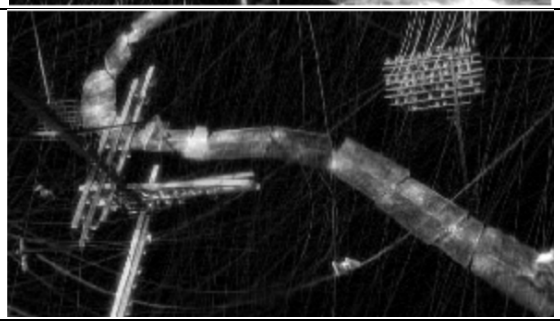
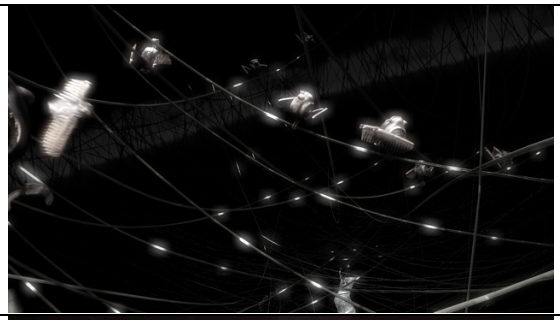
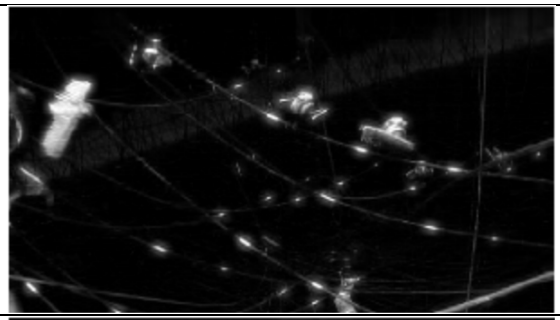
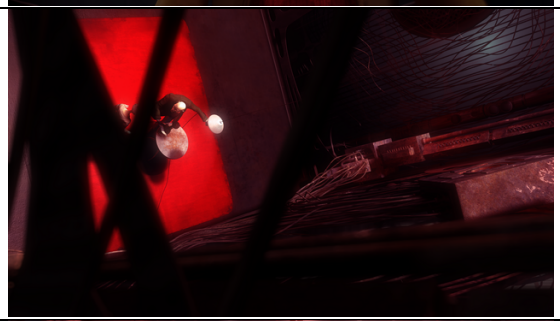
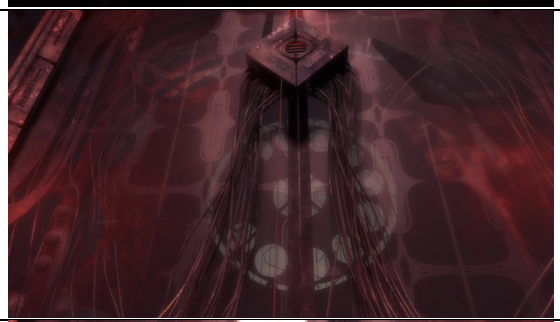
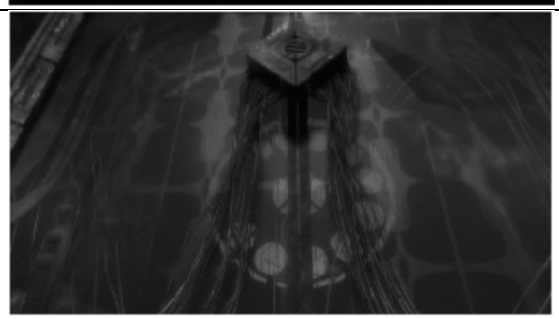
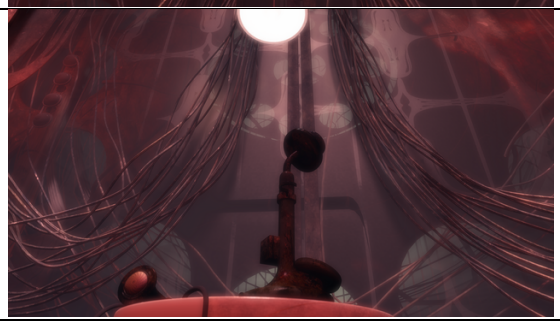
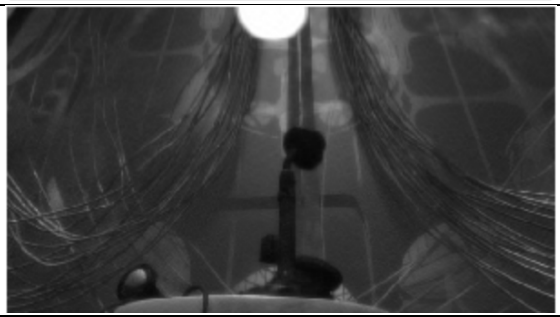
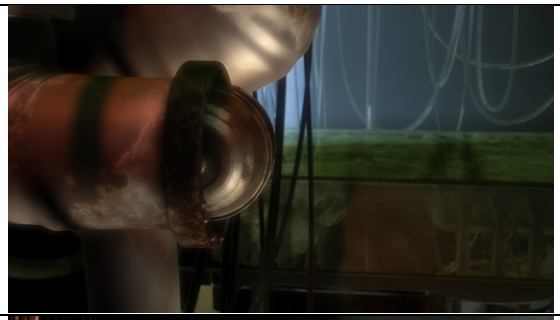
NFM Experiment 2 with Images from Elephants Dream

July 26, 2017

	Original Image	Predicted Image (d_cols = 100)	Norm Difference
1			0.2522
2			0.6472
3			0.7380
4			0.6669
5			0.3491

6			**0.2198**
7			**2.3602**
8			**3.4344**
9			**5.2401**
10			**2.7894**
11			**3.4728**

12			3.5649
13			0.3875
14			0.5230
15			1.5515
16			0.7455
17			0.8234

18			**0.5225**
19			**1.2629**
20			**0.7631**

OBSERVATION:

After experimenting the NFM algorithm with 20 different images from the Elephants Dream movie, I observed that even **d_cols = 100** does not work very well in some cases—as can be seen above. It seems to me that the darker the original image is, the worse results that the NFM produces. When considered, the images from 1 to 6 have many pixels that are **not** dark or close to black color—which has RGB values of (0,0,0). Accordingly, the **predicted** images for them are close to the original images and the norm differences are low. Nonetheless, the images from 7 to 12 have predominantly blackish pixels; therefore, the **predicted** images for them are not as good as the ones for the first six images.

In conclusion, I think that this is happening because the individual **Amat** matrices for dark images have a lot of **zeros** AND/OR numbers that are close to zero—because colors that are close to **BLACK** have RGB values that are close to **zero**. This "sparse" property makes it harder for our algorithm to find good **X** and **Y** matrices that satisfies **Amat = X * Y**—even when **d_cols = 100**.