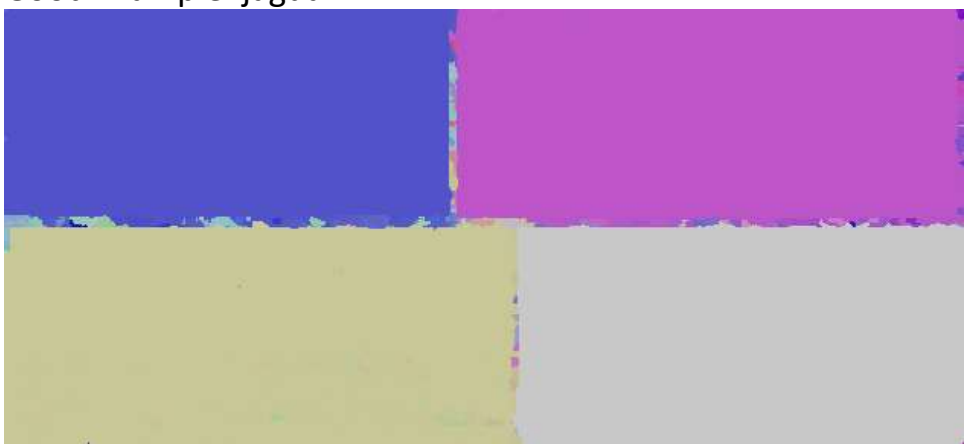
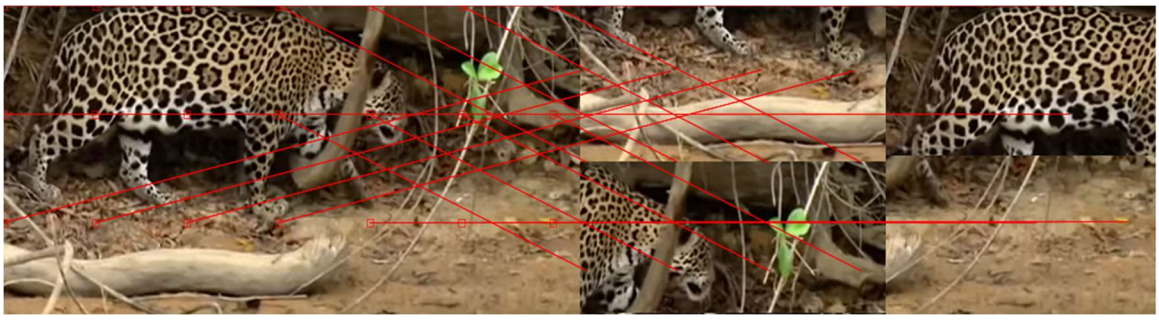
This is the report for Assignment 3. I ran the scripts on photos provided, searched online or taken by my phone.
It's clear to see that after iteration 2, most of the nnf will not change anymore, as they've already find their local minimum.

If some patches, shapes, colors or others not existed in target, the PatchMatch algorithm will try to find a closest one from other parts to pretend it's correct. However, it's easy to be identified by human.
PatchMatch might be misled when some parts (in a large size) are highly similar with other parts. As a result, that part would be highly blurred.
For most of the situations, PatchMatch will try to fill in the missing part, when ever, the image is zoomed-in, shape changed, rotated and the like, or even the object itself is moving. However, if there is enough element for the algorithm, Patch Match will try to fill in a possible part, even sometimes this fix is not exactly accurate and those parts are highly blurred.

The followings are the performance for these circumstances.

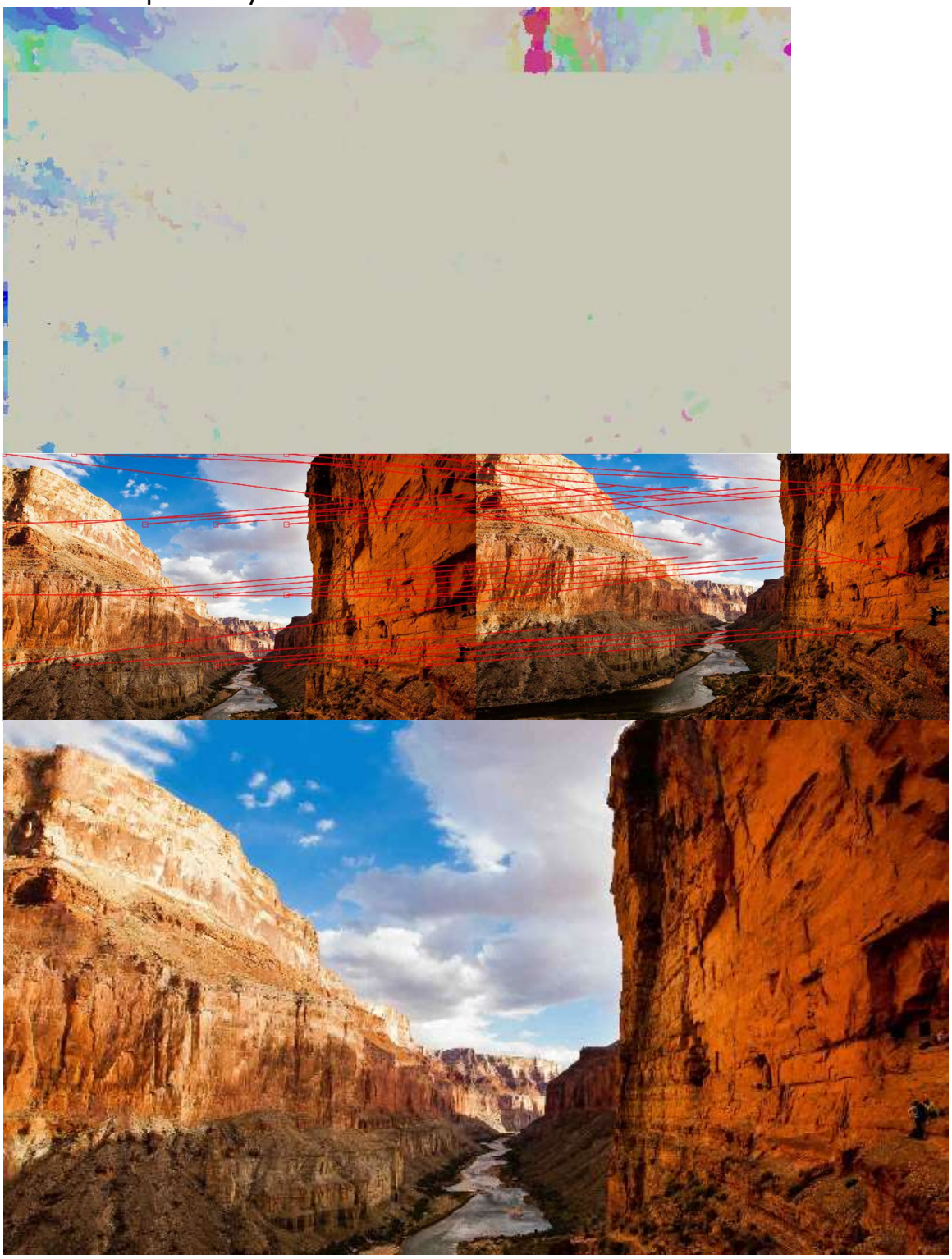
Good Example: jaguar2





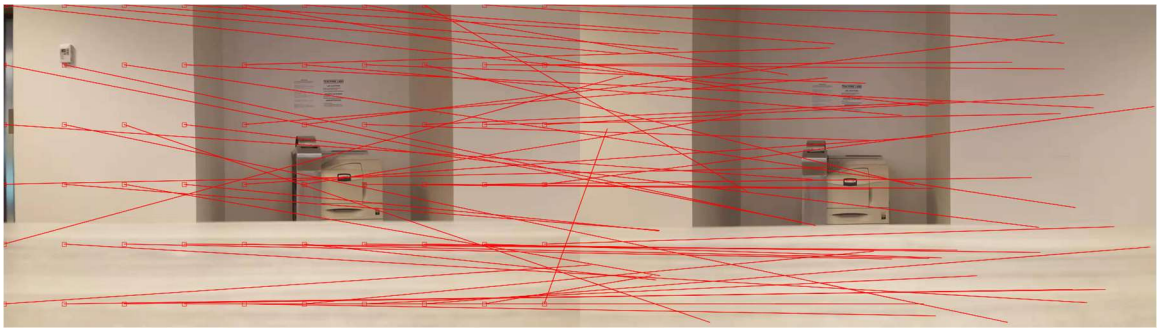
This is a good example. The target image is just the reorder the cut of source pictures. Thus, all information from target and source are matched, although it's in different position. That's the reason we can see pure color block in nnf image.
We can see that the blur on src reconstructed is very light.

Good example: Canyon



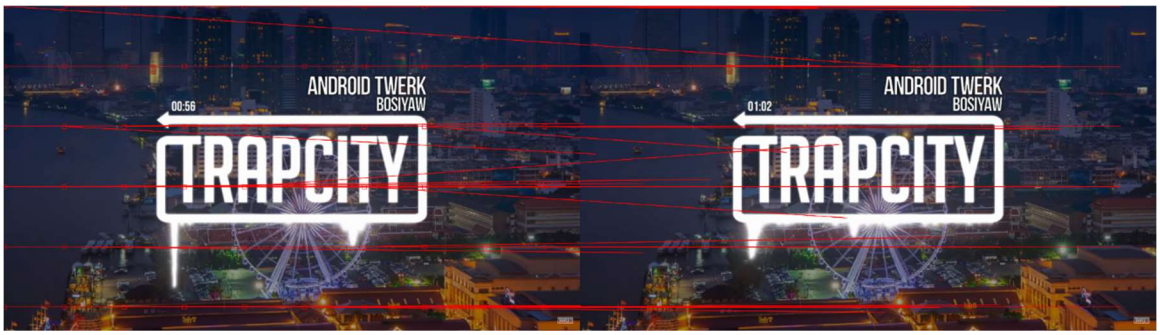
We can easily find that the target image is just a parallel translation from the source image. The camera moves a little bit downwards. Most of the information already exists in target, which is almost blank in nnf. As the high part of sky in source file is not existed in target, it is reconstructed using other information. It blurred compared with the source, as you can see the cloud at the top left. However, the total result is still very clear and good.

Passable result: BA3175



This is a result for "stereo images". The point is that, the target doesn't contain the information about the controller on the wall and the door information so they are constructed with other information. Thus, they are blurred. The desk is white with some pattern. The parts on desk are similar, so it's confusing for the algorithm to select information. Thus, the desk is blurred.
However, the printer is extremely clear. All information for the printer is included and specific to find. The blur on printer is very light. It's also easy to find on nnf image that the printer part and the bulletin above are very light, which mean they find the information on specific target for these.

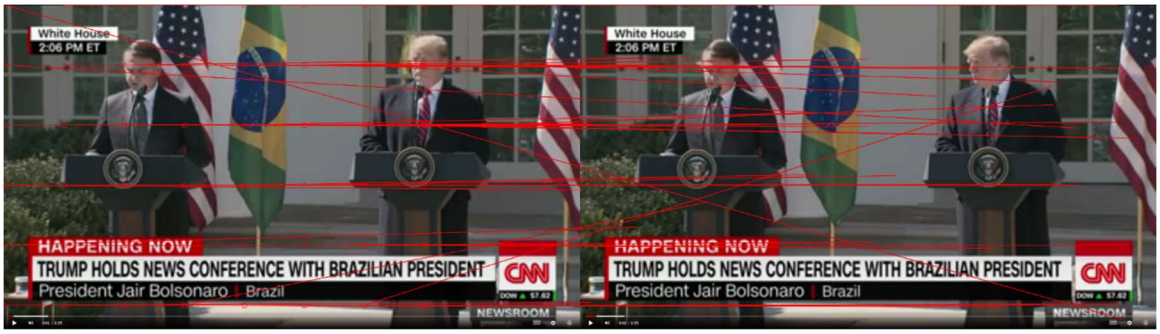
Passable Example: Android Twerk: [Bosiyaw - Android Twerk]



These are two frames at 00:56 and 01:02. Most of the background could be found easily from the target as nnf is almost blank . There are two parts have some problems:

1. The wave crest under word "TRAPCITY". The target doesn't have such a long crest so it's constructed from other place. It's blurred so it doesn't seem natural.
2. The time above the arrow. The script failed to construct the 56 as it doesn't have a shape for "5" and "6". The position of "6" is too close to the building in the background so it's hard to find patch to compose "6" and its background. Thus, it looks like "00:60".

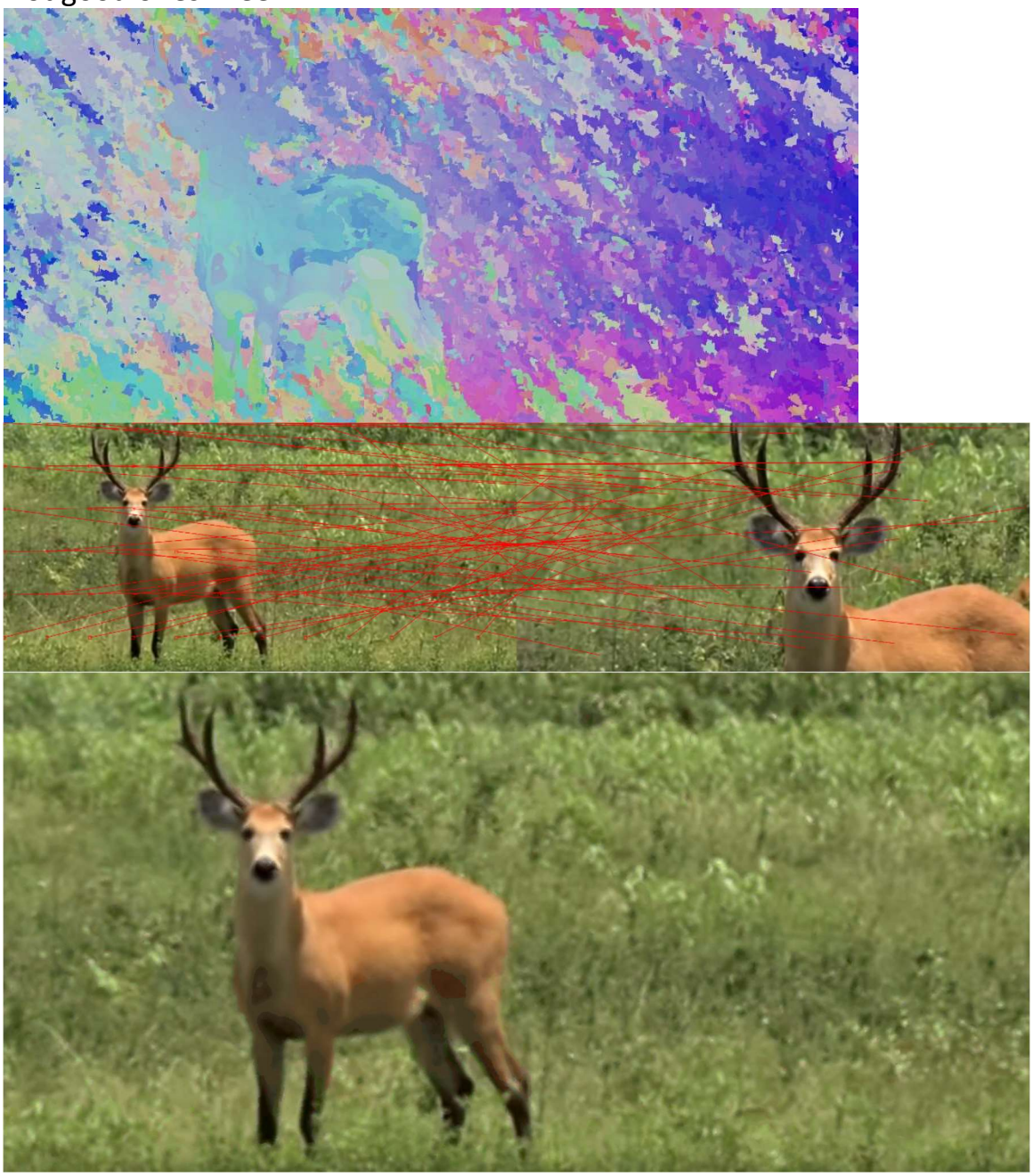
Passable example: Trump



https://www.cnn.com/videos/politics/2019/03/19/trump-bolsonaro-fake-news-bts-nr-vpx.cnn

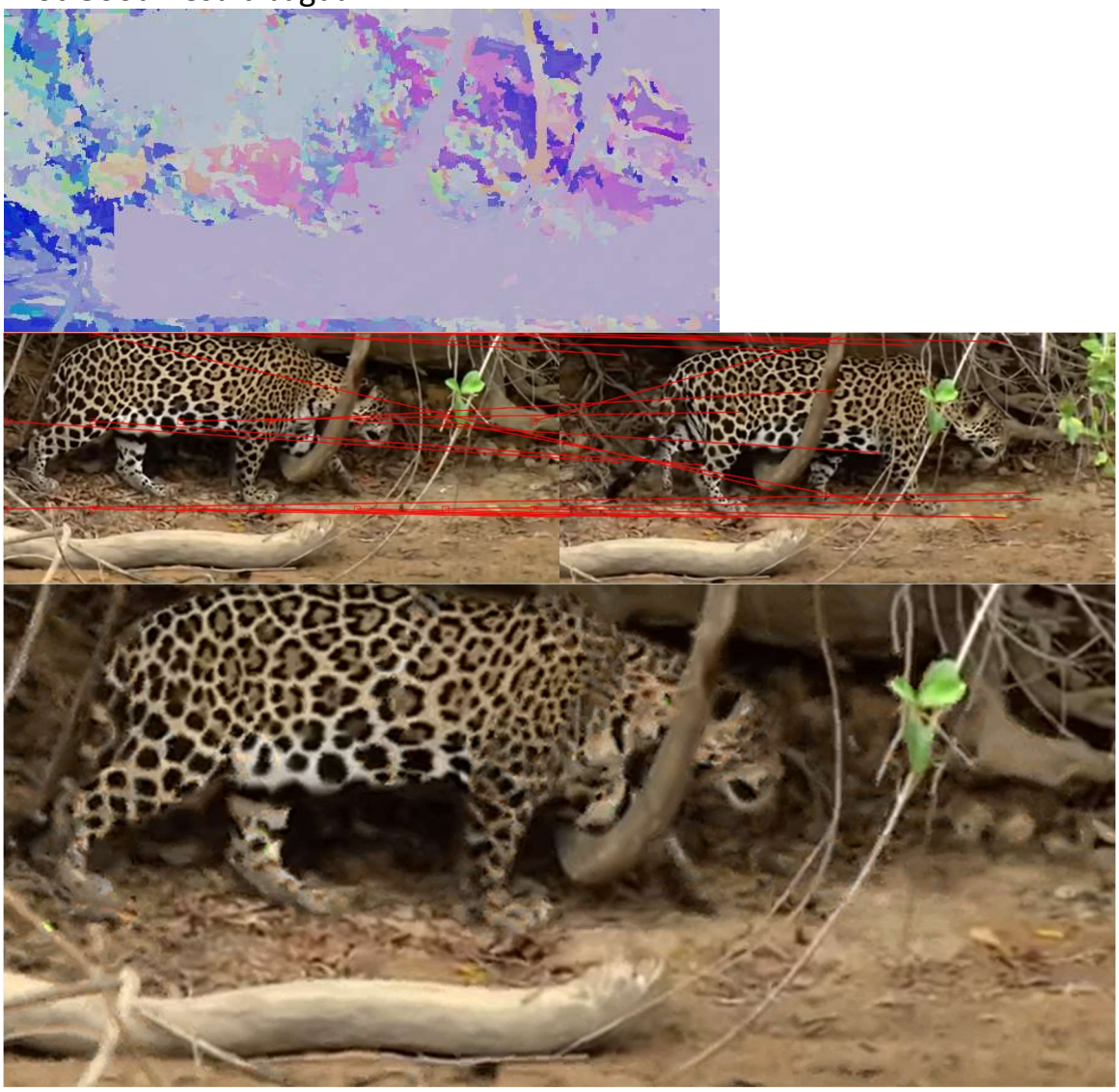
As from nnf, most of the background and the president of Brazil are the same, so they are blank.
Donald Trump threw his eye on Jair Bolsonaro, so there is no information for his right face. This part is constructed from other patches, so Trump's face seems blurred and unnatural. Similar problem for his tie. In target part, the right part of his tie is covered by microphone but not in source file, which is very bright. Thus this part need to be constructed from other parts of ties and it's hard to find such information. That's why it seems blurred there.
And also the time at the left bottom. The source time is 0:01, when there is no information of "1" in target space. It's very hard to construct it with the status bar. Thus a "2" is placed at that place, which is the most similar one compared with "1".

Not good ones: Deer



The target is actually a zoom-in version to the source.
The deer has similar color on it's body and the background is almost green. However, it's still a kind of mess on nnf image. While the target only contains the deer horn and upper body, it's very hard to construct the patches for it's lower body and leg. We can see that some of the points from the leg point to the horn, as the leg is closer to the black color. The lower body and the legs are very strange and they are highly blurred.

Not Good Result: Jaguar

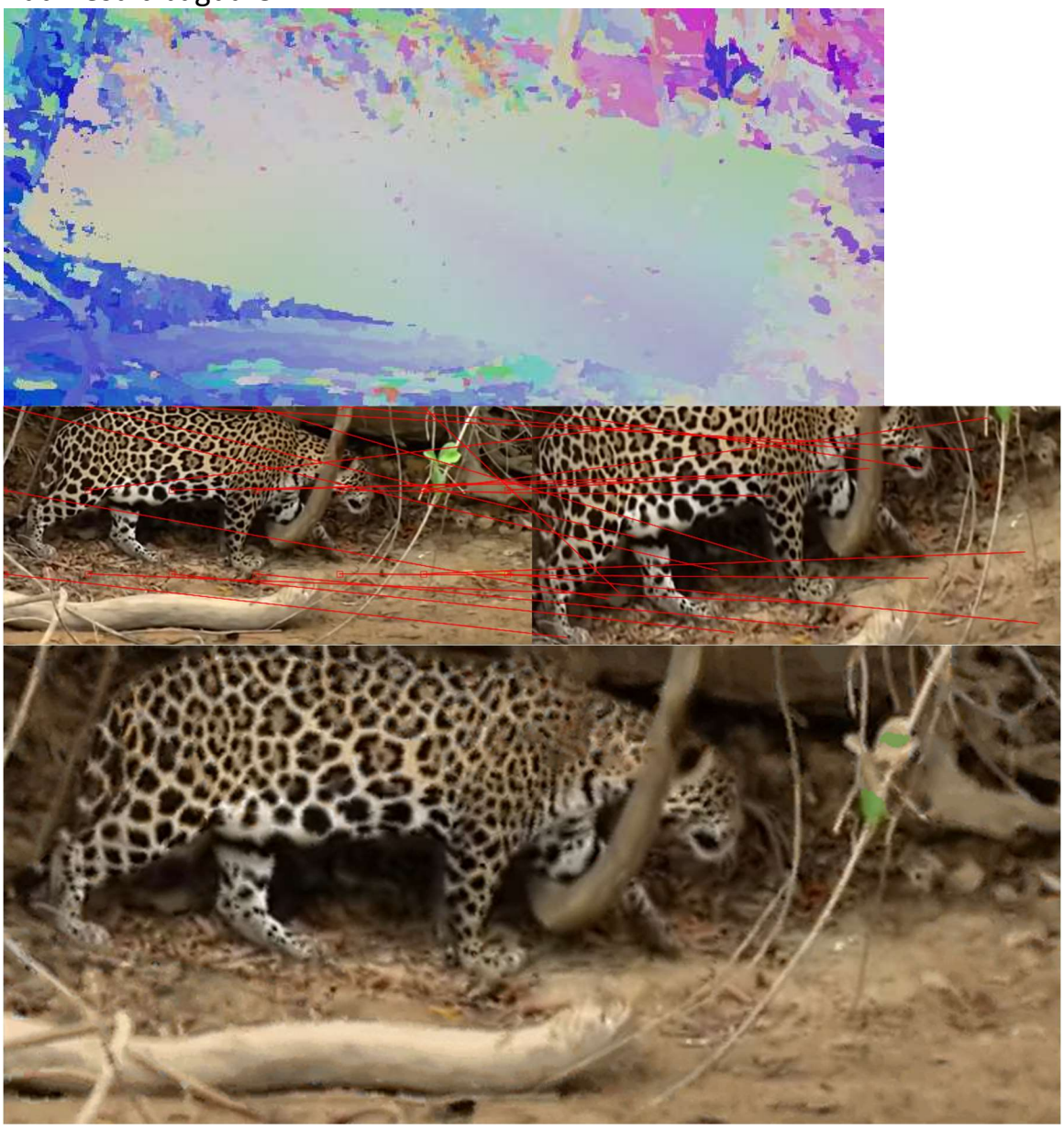


It's clear to see the jaguar moved. In target file the leaves cover part of the neck of the jaguar and waist. It's easy to fill in the waist part, as it can get information nearby(the stick shape on nnf at jaguar waist but very tint). However, it's hard to find the information of the neck part and it's clear to find it strange.

It's also hard to find information about "twigs in the shadow" in source as most of the twigs are bright and clear in the target image. Thus, it's highly blurred here.

If it's able to see it clearly, there are strange pixels around the clover. These are pixels from the jaguar. The reason is that the clover is covering the jaguar in target picture and some jaguar pixel information is also carried to the result.

Bad Result: Jaguar3



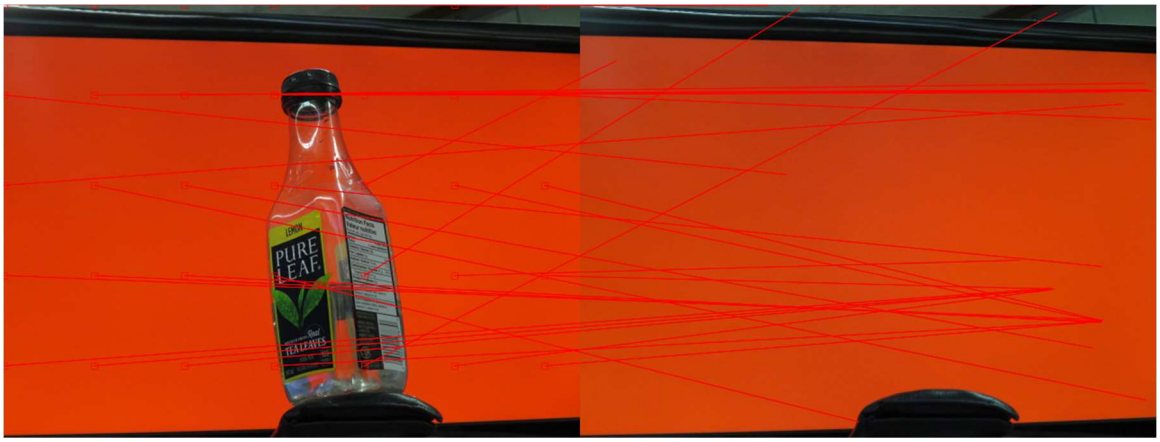
The tiger is still moving. But it can be seen as a rotation and zoom-in from the source.

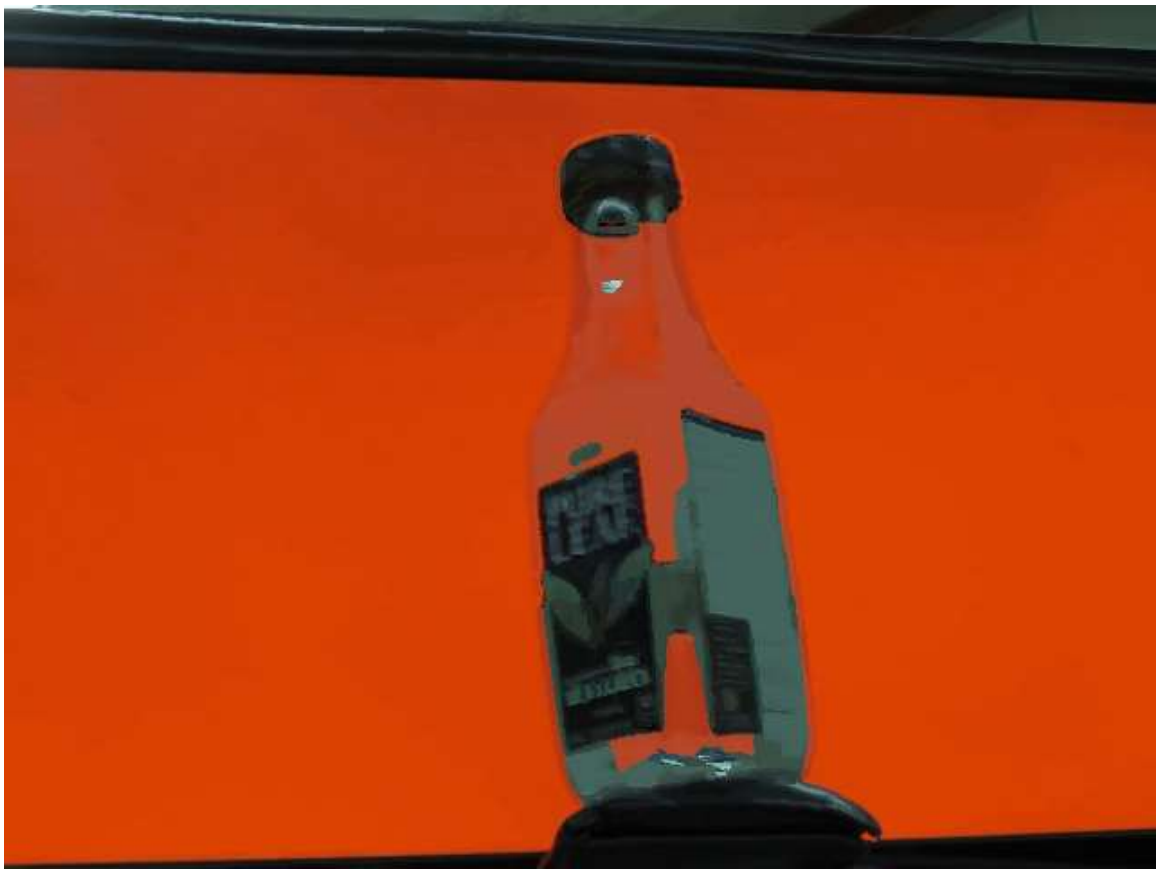
In NNF, it's clear to find a blank ,rotated rectangular, that's what is matched to the target file. This is an evidence to say the target is a kind of rotation.

The most problem in this picture is the color of the clover. The algorithm failed to paint it green, as it's almost impossible to find patches for green clover in target. Thus, the algorithm will find something similar. However, it's easy for human to find out the strange clover.

And the target is blurred than source, so the result is more blurred than source.

Bad result: bottle





This is a bottle before the red screen. However, there is no such bottle in target file!

We can clearly see the edges and the wallet matched correctly and blank in NNF. The red screen matched randomly with color difference of red. It's not totally random.

However, as there is no such bottle in target, all it's information must be borrowed from somewhere else. From NNF, we can clearly see the bottle in color which means all it's information must be borrowed. We can see the color is totally different than the source and the label is broken. (As some part of the label is closer to the screen, compared with the edge background.)