3D Computer Vision Project 4

Bodong Zhang

Practical Problem

3 Optical Flow

Introduction:

Optical Flow mainly use two images with short time difference to deduce velocity of object in image. The assumption is that the same point in object remains same intensity. Also, the image is smooth enough so that we can get accurate derivation.

Method:

Since intensity remains same, dI(x,y,t)/dt=0, so (∂I/∂x)*u+(∂I/∂y)*v+(∂I/∂t)=0. Because there is only one equation but two unknown vectors, so we can only get normal vector(perpendicular to straight line) without any other thoughts. Also, we can combine neighbor pixels and assume they have same velocity to increase the sum of equations so that we can calculate the true velocity.

Experiment:

1 First choose two consecutive images toy_formatted2 and toy_formatted3

toy_formatted2

toy_formatted3

We can observe that vehicles are moving on the ground but the location of camera is also moving, thus the ground has also non-zero velocity.

2 Apply smoothing to images

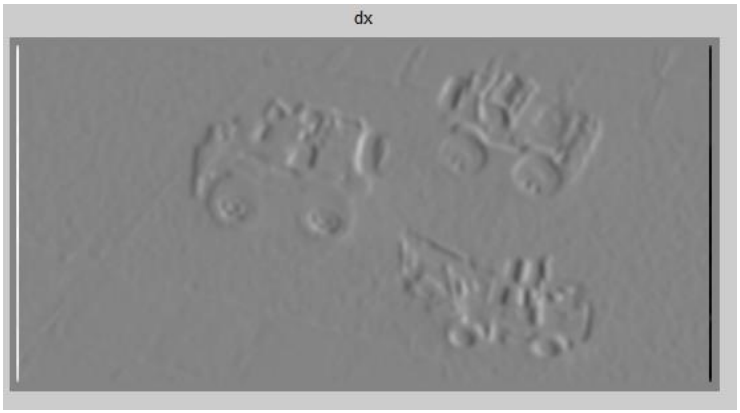
We use Gaussian filter,

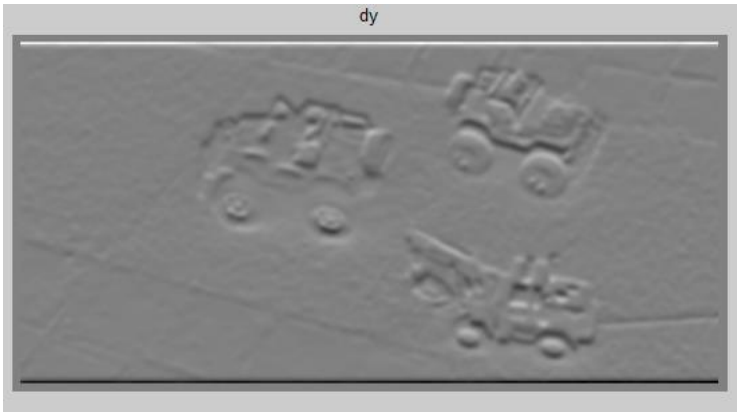
toy_formatted2

toy_formatted3

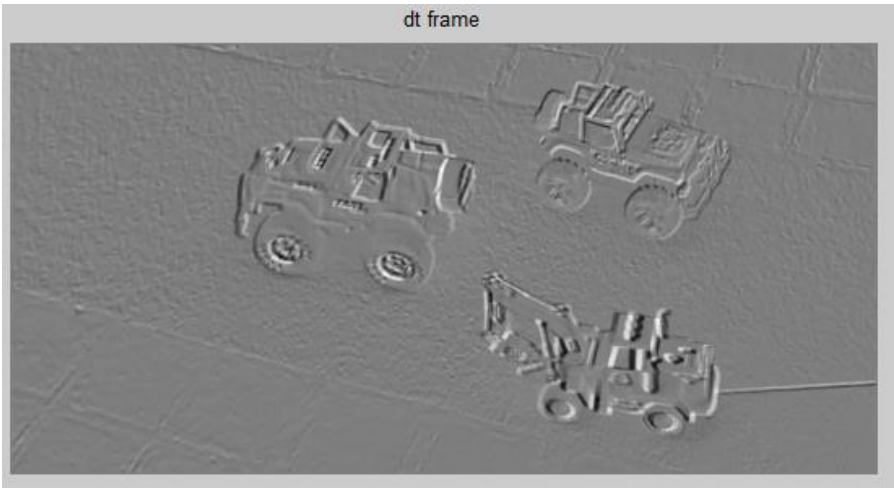
3 calculate dI/dt by using blurred images

4 calculate dI/dx, dI/dy





5 Calculate the time gradient by the difference between consecutive frames



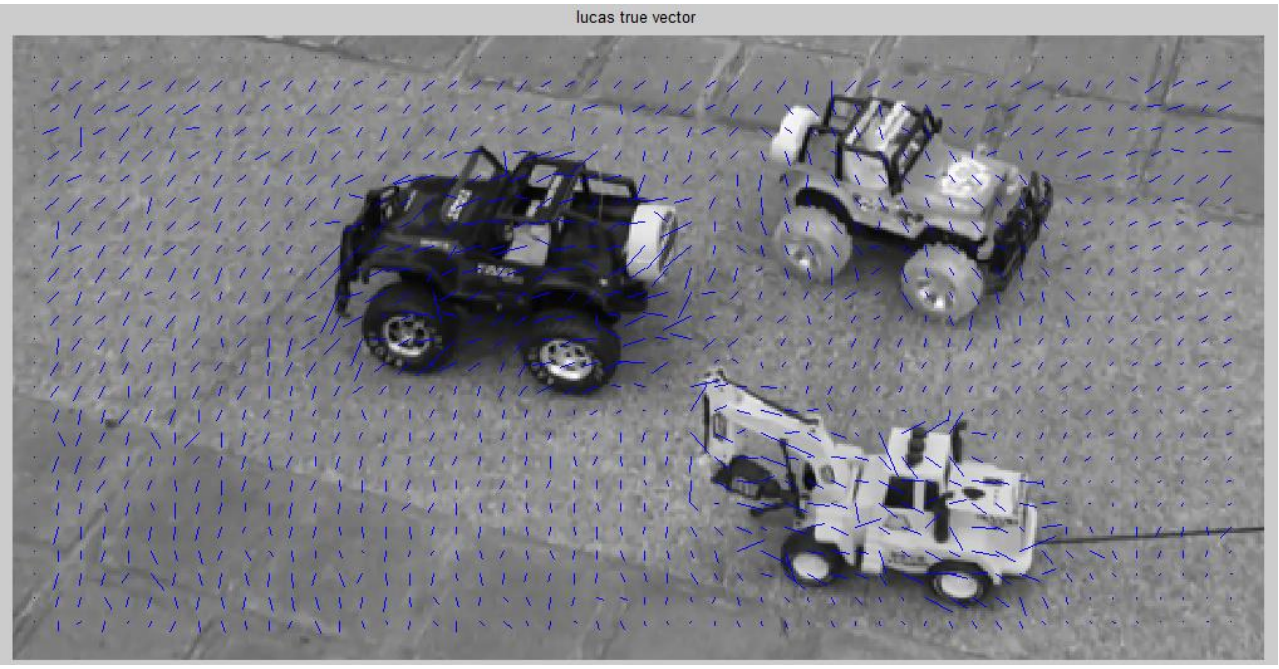
Then we can first calculate normal velocity and show it on image.

We can see that the velocity is perpendicular to boundaries(like near tires). And near the vehicles the norm of vector is large. But the direction is chaotic. So next we combine neighbor pixels to get real velocity.

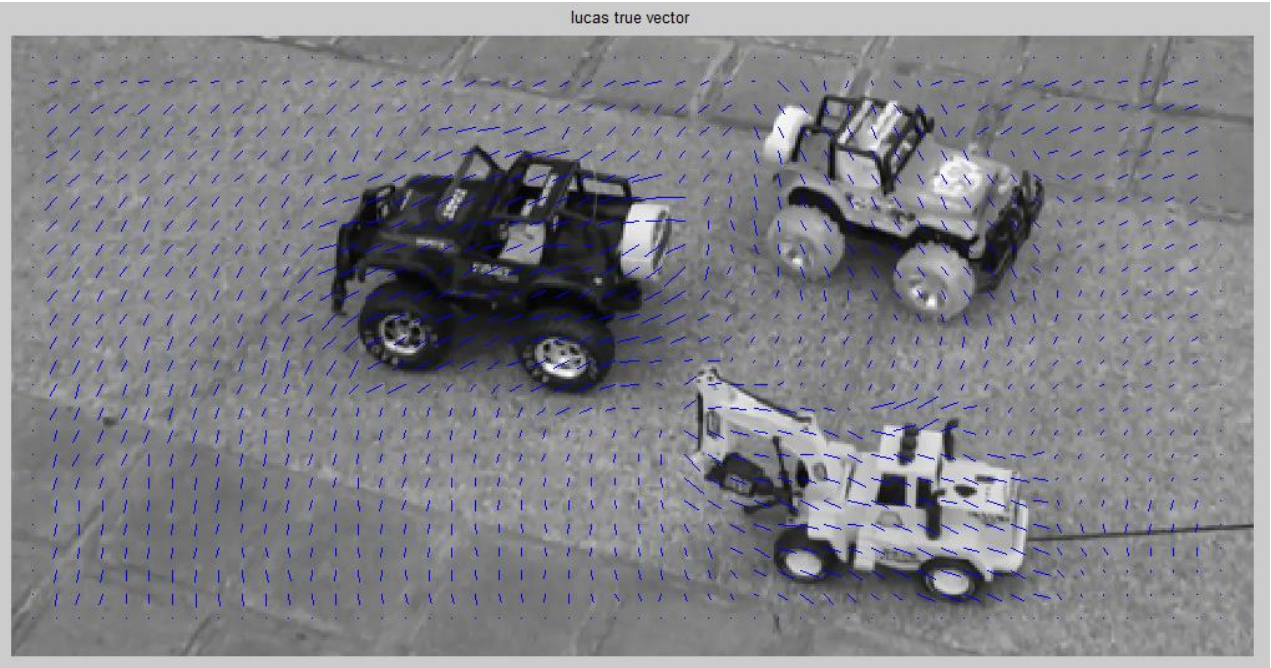
We can vaguely see that the vectors near vehicles are horizontal, but the results are still a little chaotic.

4 Bonus: Lucas and Kanade Method

The main problem of above method is that it is calculated based on pixel rather than patch, which makes it sensitive to noise. By experiment, if we use Lucas and Kanade method and enlarge size of patch, the result would be better.



Patch size 9*9

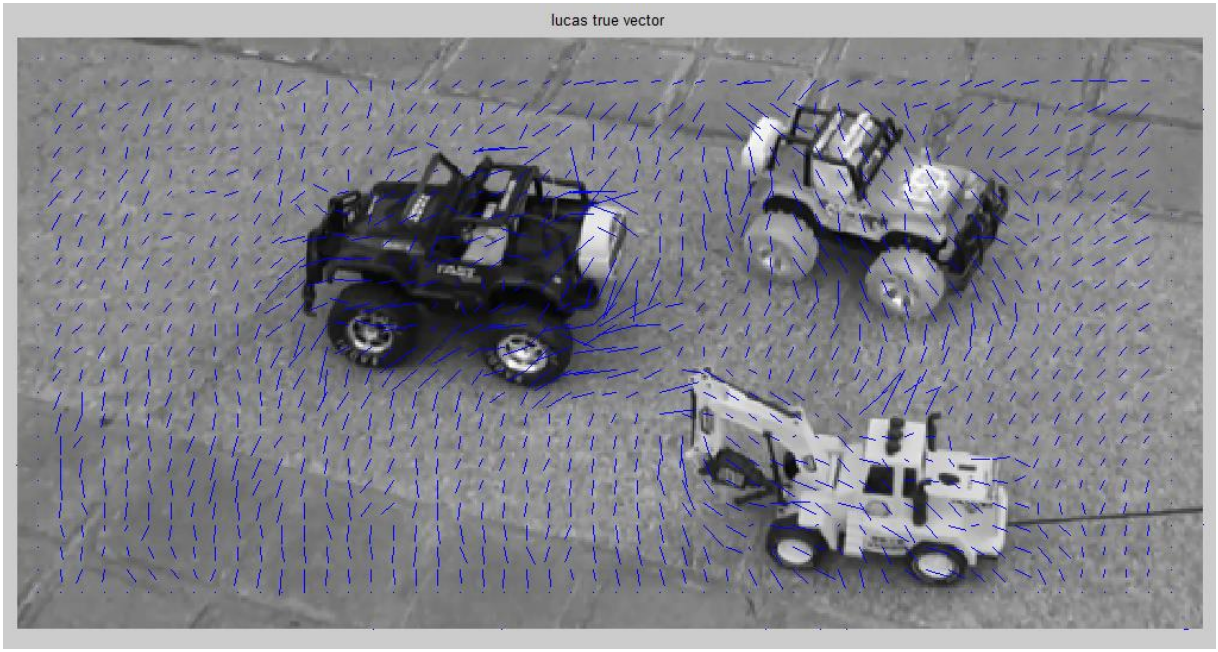


Patch size 19*19

By enlarging the size, the vectors are more close to real direction (horizontal). We can also see that the ground is rotating by bottom right corner, so velocity vector is perpendicular to radius.

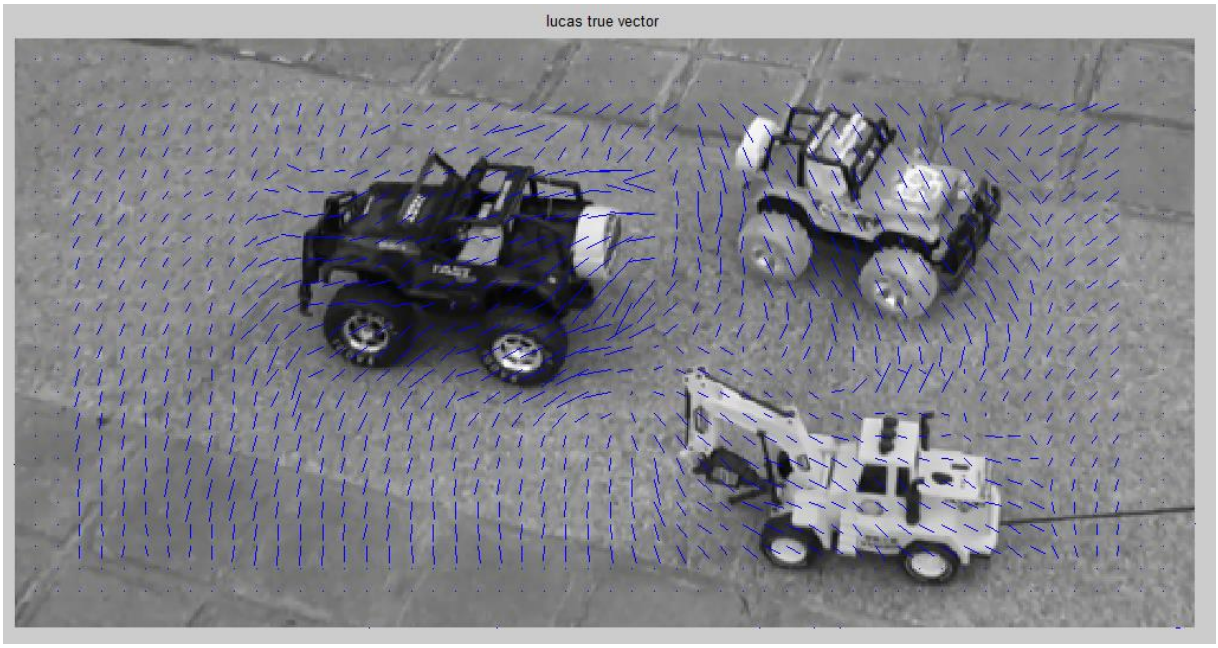
Now we run the program on other sequences.

toy_formatted3 and toy_formatted4 Patch size 19*19



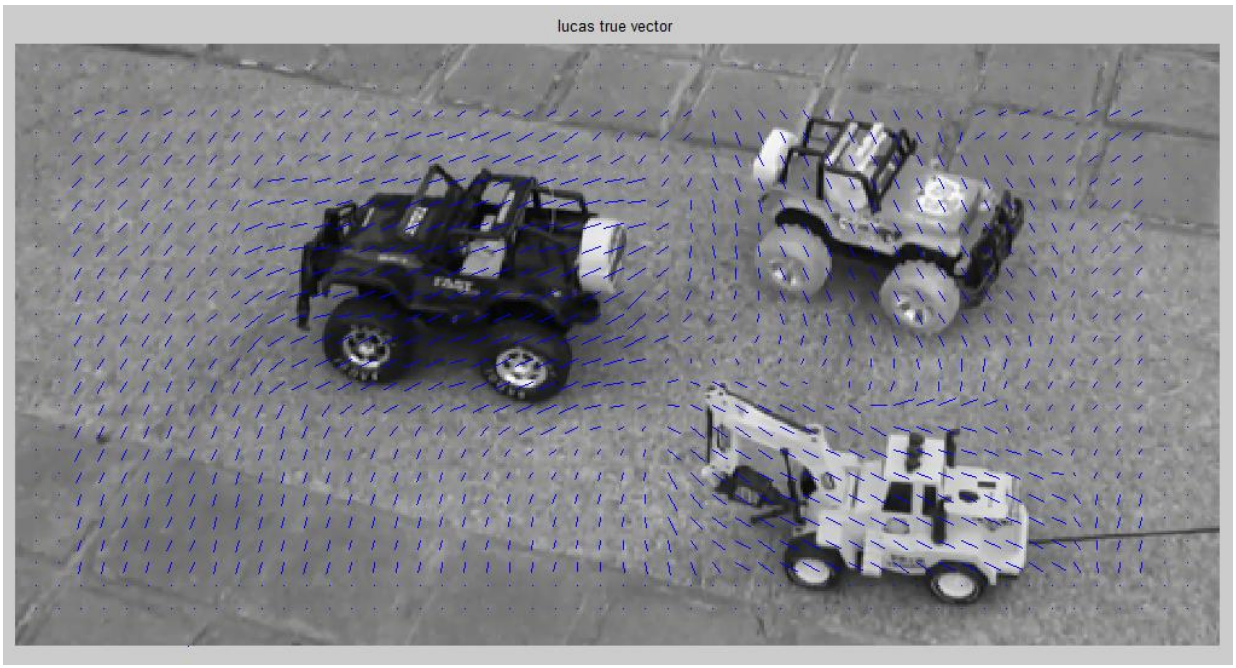
toy_formatted4 and toy_formatted5　　Patch size 19*19

toy_formatted4 and toy_formatted5 have large velocity, so we can try to use larger patch size.

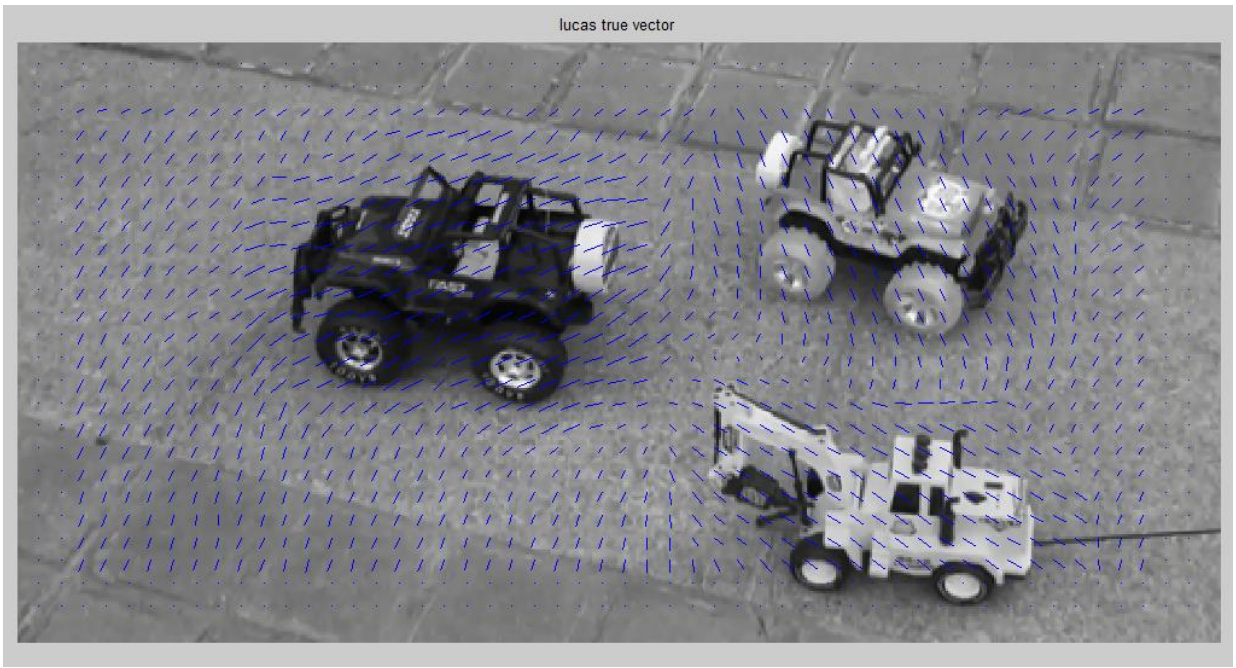


toy_formatted4 and toy_formatted5　　Patch size 41*41

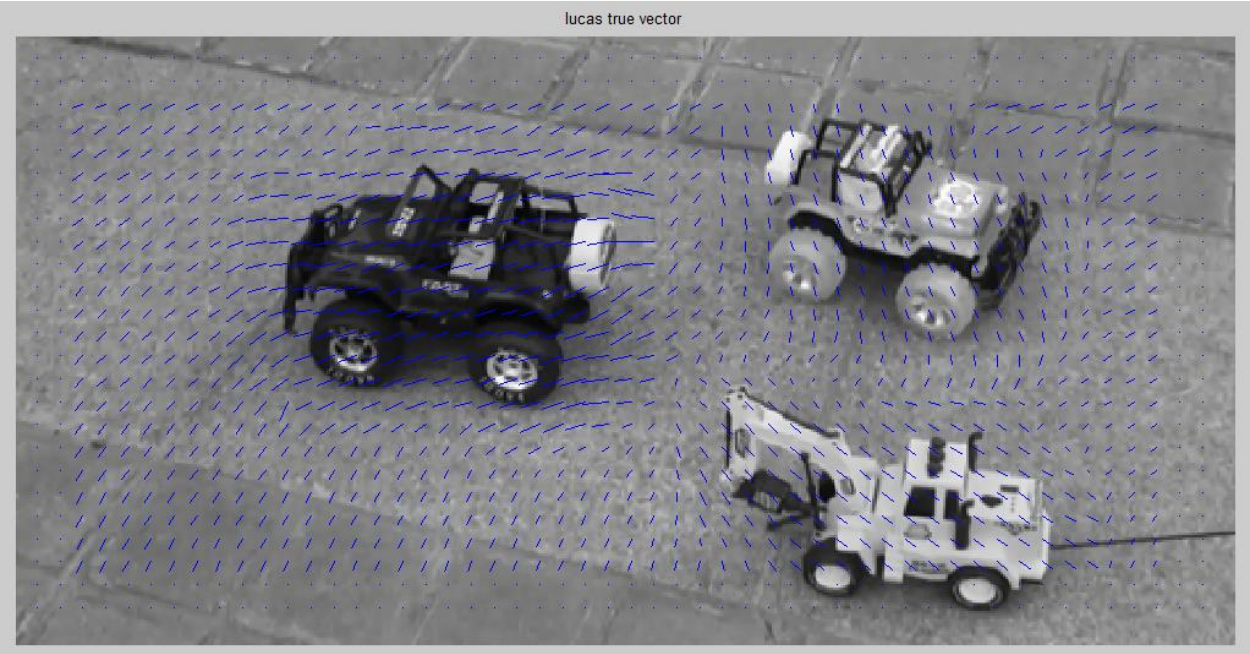
The vectors become more regular.



toy_formatted5 and toy_formatted6 Patch size 41*41



toy_formatted6 and toy_formatted7 Patch size 41*41



toy_formatted7 and toy_formatted8 Patch size 41*41

toy_formatted8 and toy_formatted9 Patch size 41*41

Conclusion:

In this project, optical flow algorithm is tested and may not have good results because of high velocity or noise. Thus Lucas and Kanade method is adopted and use patch as basic unit, which produces relatively good result.