Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Владимирский государственный университет

имени Александра Григорьевича и Николая Григорьевича Столетовых»

(ВлГУ)

Колледж инновационных технологий и предпринимательства

**Лабораторная работа №3**

на тему: «Создание видеофайла с движением объекта»

Выполнила:

Ст.гр. ИРсп-121

Сорокина Е. М

Проверил:

Хотак М. А

Владимир 2023

**Цель работы:** Познакомиться с технологией создание видеофайлов содержащих движение объектов.

**Ход работы**

**На 4**

**Задание 1.** Движение мяча по горизонту

Программный код

import numpy as np

from PIL import Image, ImageDraw

import cv2

videoDimensions = (1280, 720)

videoFps = 60

videoCodec = "avc1"

videoFilename = "poGorizonty.mp4"

videoLength = 5

videoFourcc = cv2.VideoWriter\_fourcc(\*videoCodec)

video = cv2.VideoWriter(videoFilename, videoFourcc, videoFps, videoDimensions)

img = Image.new('RGB', videoDimensions, color="black")

imgDrawer = ImageDraw.Draw(img)

imgDrawer.line([(0, 360), (1280, 360)], fill="red", width=1)

totalPercent = 0

for frameNumber in range(0, videoFps\*videoLength):

frame = img.copy()

shift = (frameNumber / videoFps) \* (1330 / videoLength)

frameDrawer = ImageDraw. Draw(frame)

frameDrawer.ellipse([(-50 + shift, 310), (shift, 360)], fill="red")

video.write(cv2.cvtColor(np.array(frame), cv2.COLOR\_RGB2BGR))

currentPercent = round(frameNumber / (videoFps\*videoLength)\*100)

if currentPercent > totalPercent:

totalPercent = currentPercent

print("%d%% has been finished" % currentPercent)

video.release()

print("Done")

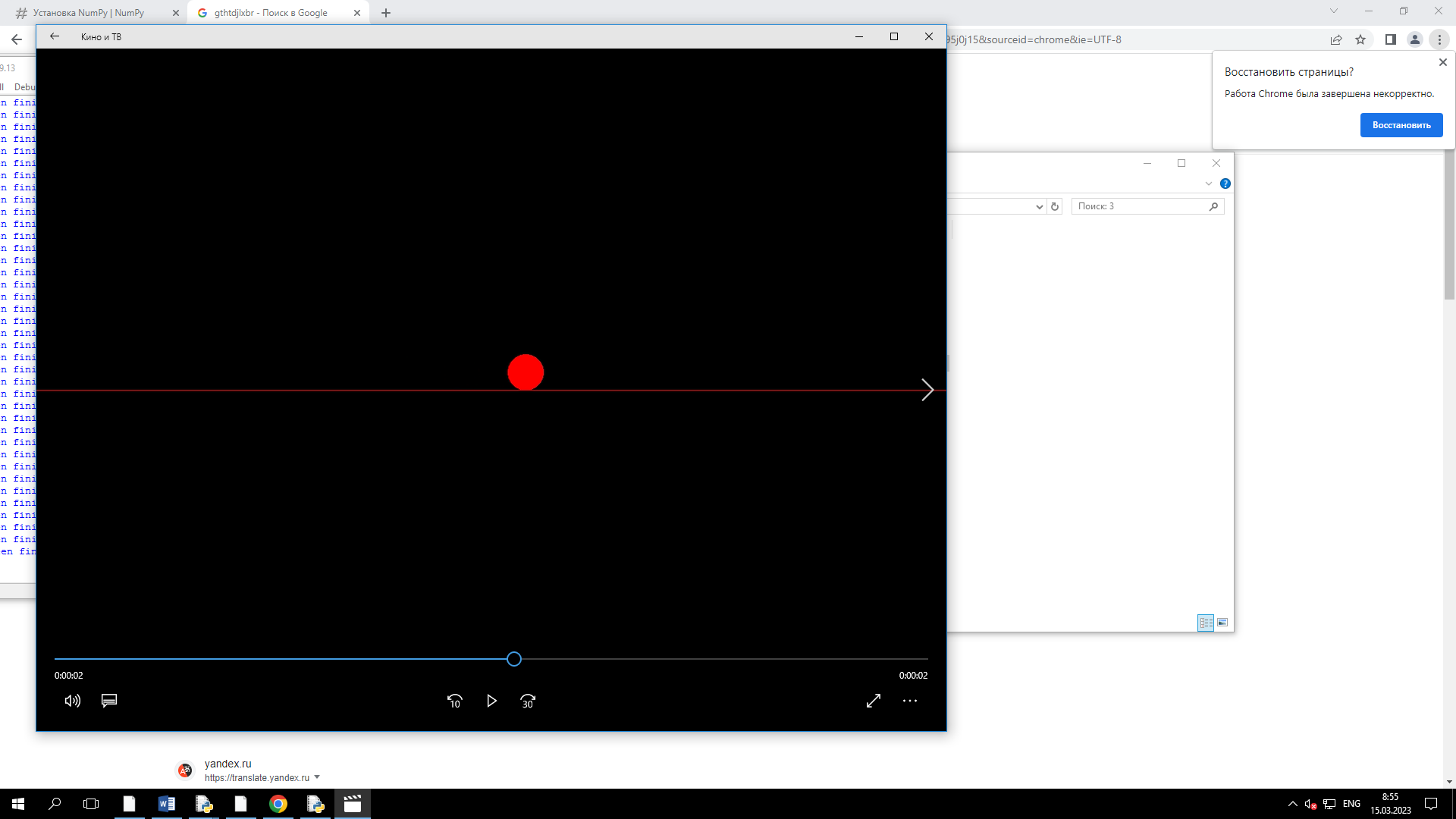


Рисунок 1 - Результат

**Задание 2.** Программа падения мяча с ускорением

Программный код

import numpy as np

from PIL import Image, ImageDraw

import cv2

videoDimensions = (1280, 720)

videoFps = 60

videoCodec = "avc1"

videoFilename = "yskorenie.mp4"

videoLength = 5

videoFourcc = cv2.VideoWriter\_fourcc(\*videoCodec)

video = cv2.VideoWriter(videoFilename, videoFourcc, videoFps, videoDimensions)

def drawCup(drawer):

width = videoDimensions[0] / 10

height = videoDimensions[1] / 2

topLeftX = 4.5 \* (videoDimensions[0] / 10)

topLeftY = 0.5 \* (videoDimensions[1] / 2)

drawer.line([(topLeftX, topLeftY),

(topLeftX, topLeftY + height),

(topLeftX + width, topLeftY + height),

(topLeftX + width, topLeftY)],

fill="red",

width=1,

joint="curve")

def drawBall(drawer, time):

width = videoDimensions[0] / 10 - 10

height = width

topLeftX = 4.5 \* (videoDimensions [0] / 10) + 5

topLeftStartY = 0.5 \* (videoDimensions [1] / 2)

maxY = 1.5 \* (videoDimensions [1] / 2) - height

dropHeight = maxY - topLeftStartY

shift = 9.8 \* time \* time

topLeftY = topLeftStartY + dropHeight / 100 \* shift

topLeftY = min(topLeftY, maxY)

drawer.ellipse([(topLeftX, topLeftY),

(topLeftX + width, topLeftY + height)],

fill="red",

outline=None,

width=0)

img = Image.new('RGB', videoDimensions, color="black")

imgDrawer = ImageDraw.Draw(img)

drawCup(imgDrawer)

totalPercent = 0

for frameNumber in range(0, videoFps\*videoLength):

frame = img.copy()

frameDrawer = ImageDraw.Draw(frame)

drawBall(frameDrawer, frameNumber / videoFps)

video.write(cv2.cvtColor(np.array(frame), cv2.COLOR\_RGB2BGR))

currentPercent = round(frameNumber / (videoFps\*videoLength) \* 100)

if currentPercent > totalPercent:

totalPercent = currentPercent

print("%d%% has been finished" % currentPercent)

video.release()

print("Done")

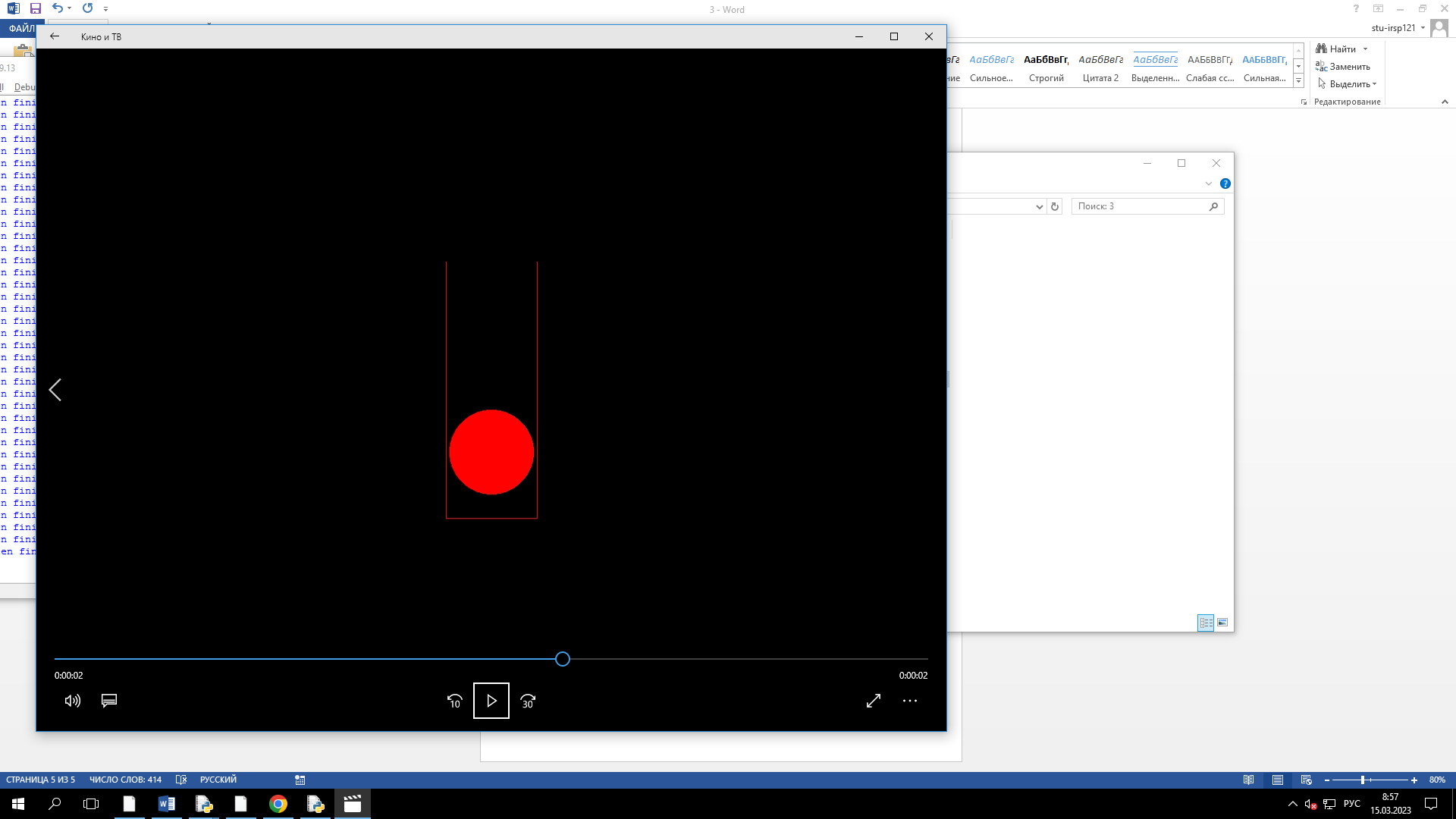


Рисунок 2 - Результат

**Вывод:** В ходе данной лабораторной работы ознакомились с технологией создание видеофайлов содержащих движение объектов.