Tracking Brownian Motion Through Video Microscopy*

Asma Khalid, Muhammad Umar Hasan,
Muhammad Hamza Humayun and Muhammad Sabieh Anwar
Center for Experimental Physics Education,
Syed Babar Ali School of Science and Engineering, LUMS

V. 2022-2; February 28, 2022

Brownian motion is the random motion of colloidal particles suspended in water, air or any other solvent. In 1905, Einstein argued that this motion is a direct evidence for the atomic nature of matter. Einstein′s and Perrin′s efforts helped raise the status of atoms from useful hypothetical objects to objects whose existence could no longer be denied.

1 Objectives

In this experiment, we will,

1. observe Brownian motion of microparticles,
2. calibrate a compound microscope,
3. use some basic and simple routines for image processing,
4. plot Brownian motion in 2-D, and
5. observe how the mean square displacement of particles can be used to estimate Boltzmann's and Avogadro's constants.

2 Theoretical Introduction

In 1827, Robert Brown observed the random motion of micro-particles suspended in gases and liquids. This jiggling motion, aptly, was coined "Brownian motion". However, it was only in 1905 that Einstein first explained this phenomenon on the basis of the kinetic theory of molecules. Einstein, quantitatively, related the parameters employed in kinetic theory such as such as viscosity and mobility with the Brownian motion.

Einstein performed a statistical analysis of molecular motion and its effect on particles suspended in a liquid. As a result, he calculated the mean square displacement of these particles. He argued that an observation of this displacement would allow an exact determination of atomic dimensions and prove the existence of atoms and verify the molecular kinetic theory of heat.

Perrin, a brilliant experimentalist, performed a series of experiments in the first decade of the twentieth century, one of which depended on Einstein's calculation of the mean square displacement of suspended particles. His results confirmed Einstein's relation and thus the molecular-kinetic theory. A historical account of these discoveries can be found in the reference [1].

2.1 Brownian motion and the kinetic theory

Brownian motion can be explained using the kinetic theory of matter and the kinetic molecular theory of heat. The kinetic theory of matter posits the existence of atoms and molecules, and their constant motion due to which they elastically collide with one another. The kinetic molecular theory of heat, similarly, describes temperature as the constant motion of atoms and molecules in matter.

Brownian motion in turn is characterized by the constant and erratic movement of minute particles in a liquid or a gas is thus due to the inherently random motions of the atoms or molecules that make up the fluid in which the particles are suspended. The fluidic atoms or molecules collide with the larger suspended particles at random, making them move randomly. Einstein described that Brownian motion actually arises from the agitation of individual molecules due to the thermal energy $k_B T$ they possess at a specific temperature. The collective impact of these molecules against the suspended particle yields enough momentum to create movement of the particles.

2.2 Mathematical picture and Langevin's model

The origin of Brownian motion can be understood on the basis of the theorem of equipartition of energy [2]. Each colloidal micro-particle, possessing a mass m is free

to exhibit translational motion. The mean kinetic energy of the particle in three dimension is,

$$\frac{1}{2}mv^2 = \frac{3}{2}k_BT.$$

This energy, though small in value, leads to a measurable amplitude of vibration for a small micro-particle. It is worth noticing that in addition to the random fluctuating force, the particles also experience a drag force (frictional force) as they are pulled through the solvent.

To find a solution to the motion of the particles, we will use the **Langevin** equation for a particle of mass m and velocity $\mathbf{v}$

$$m\frac{d\mathbf{v}}{dt} = -\alpha\mathbf{v} + \mathbf{F}(t). \tag{1}$$

From Equation (1), we can see that each colloidal particle is subject to two forces, that are,

1. the random molecular bombardment $\mathbf{F}(t)$ that causes Brownian motion, and
2. the resistive force $-\alpha\mathbf{v}$, where α is the damping coefficient related to viscosity of the fluid or solvent.

In one dimension, the scalar form of Equation (1) is written as,

$$m\frac{d^2x}{dt^2} + \alpha\frac{dx}{dt} - F(t) = 0. \tag{2}$$

Multiplying both sides of the above equation by x, yields,

$$mx\frac{d^2x}{dt^2} + \alpha x\frac{dx}{dt} - xF(t) = 0. \tag{3}$$

Using the expansion of the expression $\frac{d^2}{dt^2}(x^2)$, the above equation leads to,

$$\frac{m}{2}\frac{d^2x^2}{dt^2} - m(\frac{dx}{dt})^2 + \frac{\alpha}{2}\frac{dx^2}{dt} - xF(t) = 0. \tag{4}$$

Q 1. Using chain rule, expand the derivative $\frac{d^2}{dt^2}(x^2)$ to obtain Equation (4).

Now we use the theorem of equipartition of energy to find the average energy of single particle for one degree of freedom, which is given by

$$\begin{aligned} \frac{1}{2}m\langle v^2\rangle &= \frac{1}{2}k_BT, \\ \Rightarrow \frac{m}{2}\langle(\frac{dx}{dt})^2\rangle &= \frac{1}{2}k_BT. \end{aligned} \tag{5}$$

We average Equation (4) over time and recognizing that since F is a random force, $\langle xF \rangle = \langle x \rangle \langle F \rangle = 0$. Defining $\beta = \langle \frac{dx^2}{dt} \rangle$ and substituting Equation (5) into (4), finally yields,

$$\frac{m}{2}\frac{d\beta}{dt} - k_B T + \frac{\alpha}{2}\beta = 0. \tag{6}$$

Q 2. Derive Equation (6) (refer to [4] for help).

Q 3. Show that Equation (6) is solved by,

$$\beta = \frac{2k_B T}{\alpha} + A \exp\left(\frac{-t\alpha}{m}\right), \tag{7}$$

where A is an integration constant. For a reasonably long observation time ($t \gg m/\alpha$), we may ignore the second term on the right side which can be negligibly small. Hence, integrating Equation (7) over an observation time τ, we obtain,

$$\begin{aligned} \int_0^\tau \langle \frac{dx^2}{dt} \rangle dt &\approx \int_0^\tau \frac{2k_B T}{\alpha} dt \\ \langle x^2 \rangle &\approx \frac{2k_B T \tau}{\alpha}. \end{aligned} \tag{8}$$

This equation shows that the mean square displacement depends on the temperature T, observation time t and the coefficient of viscosity α.

2.3 Spherical Particles

For spherical particles, each of radius a, Stokes's law can be used to write α,

$$\alpha = 6\pi\ \eta a,$$

where η is the viscosity of the fluid. In such a case, Equation (8) takes the form,

$$\langle x^2 \rangle = \frac{2k_B T \tau}{6\pi\ \eta a}.$$

We can also write the mean squared displacement in two dimensions,

$$\langle r^2 \rangle = \frac{4k_B T}{6\pi \eta a}\tau. \tag{9}$$

Hence, by plotting $\langle r^2 \rangle$ as a function of time, we expect a straight line through the origin whose slope can be used to obtain Boltzmann′s constant k_B. Equation (9) is traditionally written as,

$$\langle r^2 \rangle = 4DT, \tag{10}$$

where $D = k_B T/(6\pi\eta a) = k_B T/\alpha$ is the self-diffusion constant [1], [2].

Using the relation between Boltzmann's constant and molar gas constant R, we can also use Equation (9) to find Avogadros' number N_A,

$$N_A = \frac{1}{\langle r^2 \rangle} \frac{2RT}{3\pi\eta a} \tau. \tag{11}$$

Q 4. Explain the dependence of the diffusion constant D on the damping factor α.

2.4 Significance of Brownian motion

The theory of Brownian motion has come a long way since its humble beginnings in the nineteenth century. There now exist a large number of real and technologically important applications. Only two of the applications are listed below.

In electronic devices, the discussion of Brownian motion is specifically important in understanding the effects of thermal motion of electrons contributing to Johnson noise.

Researches in the field of biomedicine have shown that Brownian motion plays a critical role in the transport of enzymes and chemicals both into and out of cells in the human body. Scientists have subsequently discovered that many fundamental processes in living cells are driven by Brownian motion which also shows potential for use as probes at the nano-scale.

3 The Experiment

After mixing some polystyrene microspheres in distilled water, we observe and record their random motion as they collide with water molecules. The observation is performed with a compound microscope equipped with a digital camera. The video of these movements is then analyzed in Matlab using PhysLab's video tracking library "PhysTrack" to extract the frame by frame coordinates of the particles. These coordinates in turn are used to calculate their mean square displacements. From the slope of the graph between these displacements and time, the value of Boltzmann's constant, k_B, and diffusion constant, D, can be extracted.

3.1 Apparatus

1. `Motic` microscope `BA210`: The microscope comes with a built-in 3.0 megapixel camera and a USB interface to connect to the computer. A diagram showing its different components and controls is presented in Figure 1.

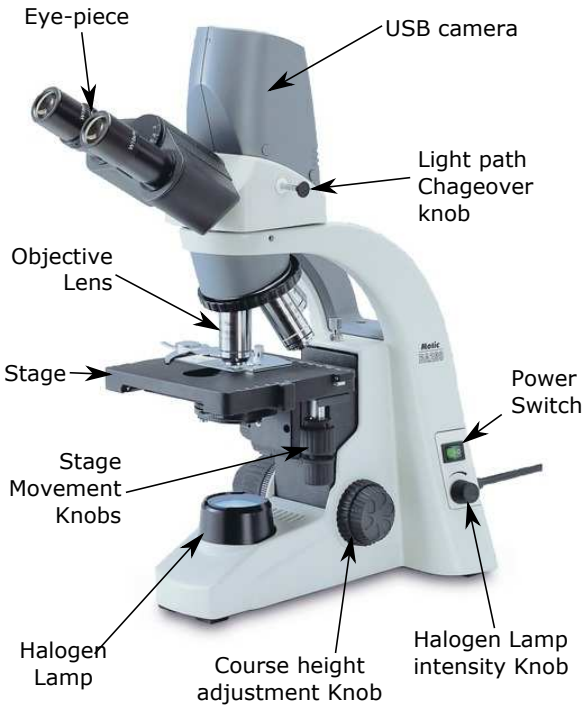

Figure 1: An illustration of the `Motic` microscope showing its parts.

2. Motic calibration slide: It has different dots and grids printed on it, along with their physical dimensions. Using this slide, a μm to pixel calibration can be performed in the microscope software.

3. 76 mm x 25 mm glass slides

4. Glass cover-slips to cover the sample so that it does not touch the objective lens or move with air currents.

5. Polystyrene microspheres (by Polysciences) diluted in water. [5].

6. 20μL micro pipette with tips.

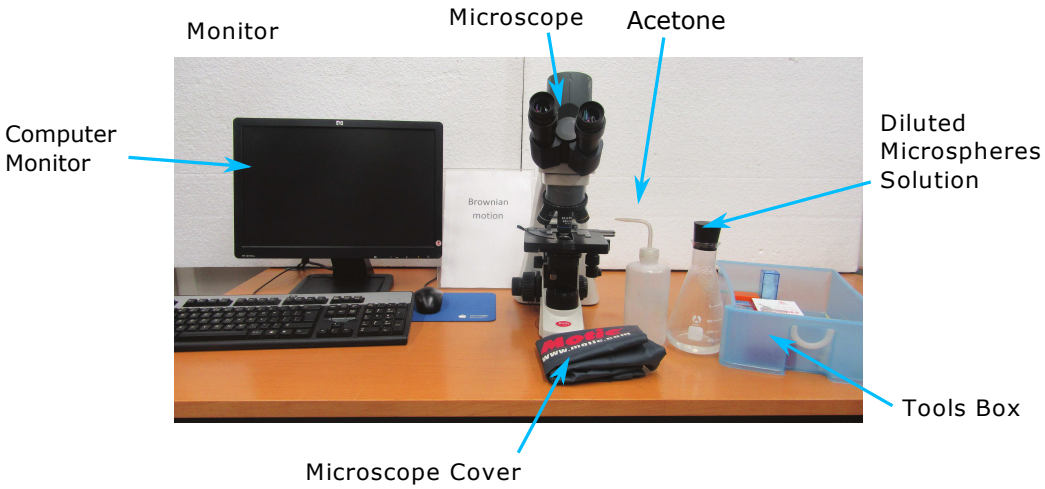

Figure 2: Setup for observing Brownian motion.

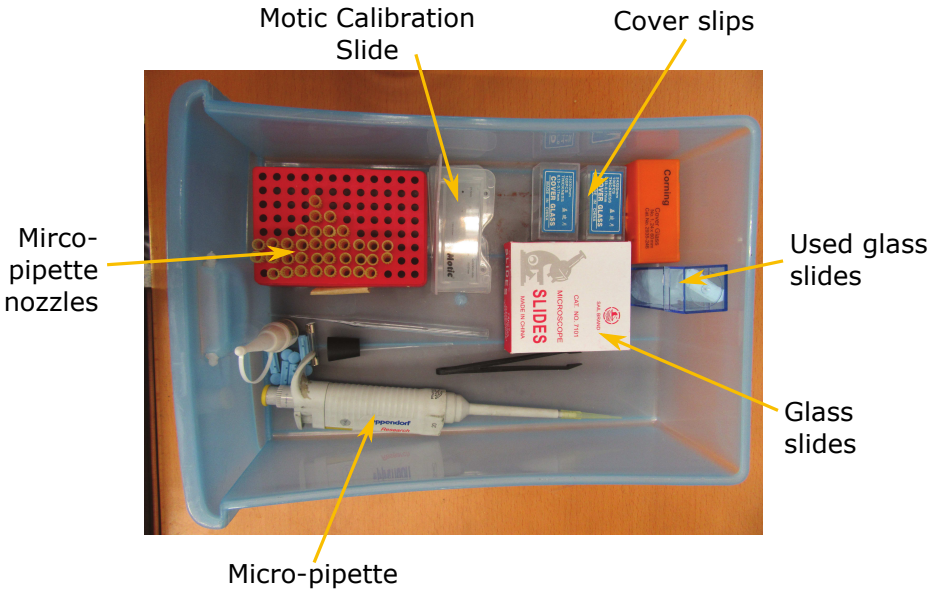

Figure 3: Tools used for preparing the samples.

3.2 Preparation of the medium

The density of particles in the packaged solution is very high, so it needs to be diluted. This increases the average distance between the particles so that they can be independently visualized and tracked. `Also remember that the microspheres should be kept refrigerated at` $4°C$.

1. Clean the slide thoroughly with water and make sure that there are no finger stains or grease marks on it.

2. Stick two pieces of plastic tape on either side of the center of the slide as shown in Figure 4. They will support the cover slip.

3. Use acetone and a cotton piece to remove any fibers or dust particles from the slide.

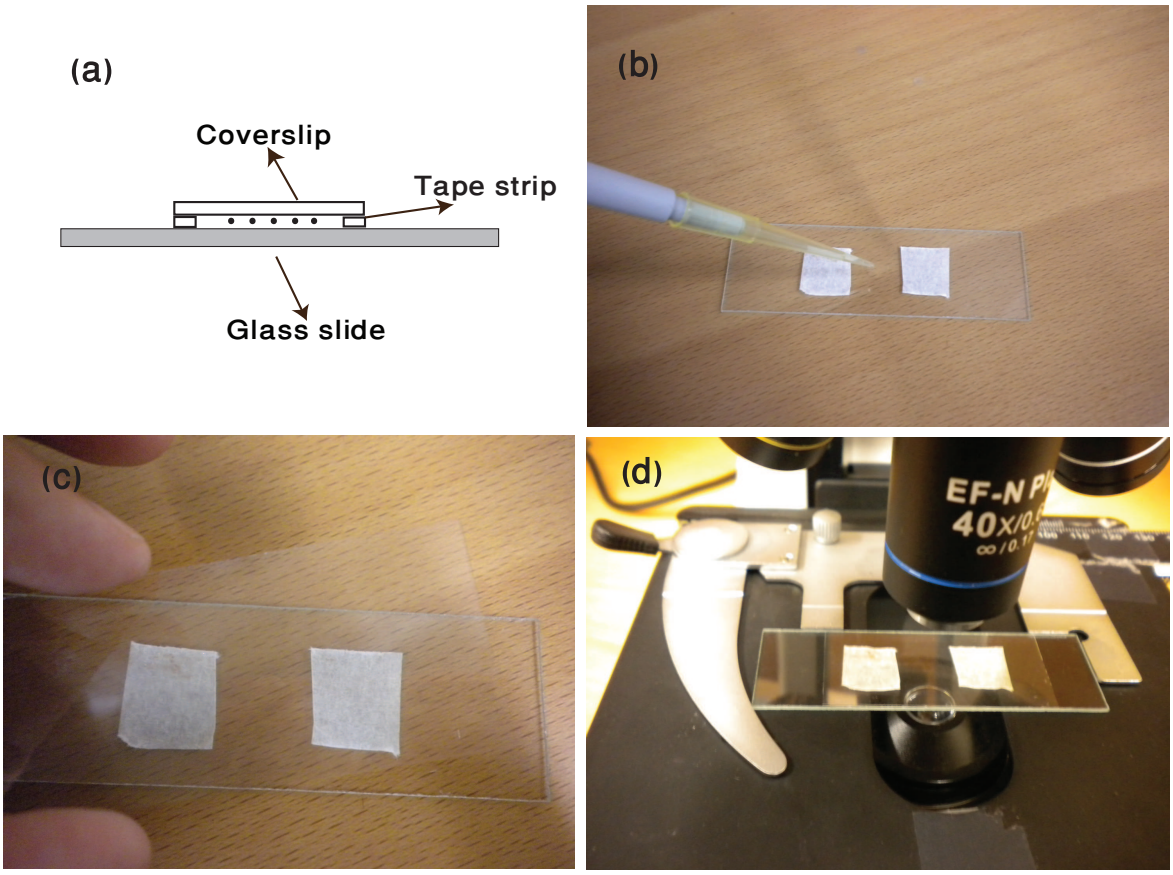

Figure 4: (a) Illustration of an observation slide, (b) using micro pipette to place the solution of a known volume at the center of the slide, (c) covering the slide with cover-slip, and (d) mounting the slide onto the specimen stage.

4. Polystyrene microspheres come packaged in a dropper bottle. Mix a drop from the dropper in 100 mL of water.

5. Set the dial of the micro pipette to 5 μL, and suck a drop of the diluted solution by gently pressing the head of the pipette.

6. Place the drop on the slide and place a cover slip, such that the drop is trapped on both sides with glass. Now place the slide under the objective of the microscope.

3.3 Setting up the microscope

The various stages of preparing the microscope for observation are now described.

1. Turn on the bulb of the microscope and set its intensity to maximum.
2. Connect the microscope to the computer through USB and run the `uc480 Viewer` application.
3. Start the application, a dialogue box asking you to choose the profile mode will appear (Figure 5). Select the `Live video` option to launch the video recording mode.

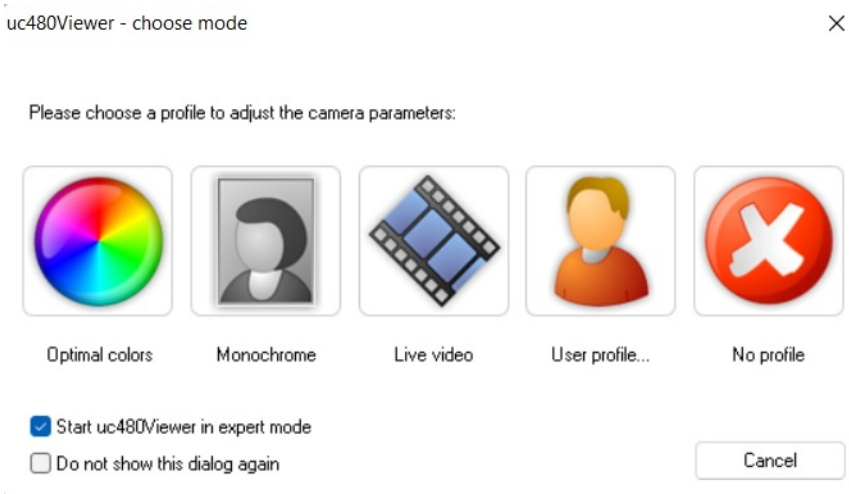

Figure 5: Start dialog with different user profile modes.

4. Select the `Open camera` option from the top toolbar to start the live mode of the application (Figure 6). If only a black window is showing, you may have to slide the optical path changeover knob to your right to allow the light to fall on the camera sensor instead of the eyepiece. Find "optical path changeover knob" in Figure 1 for explanation.

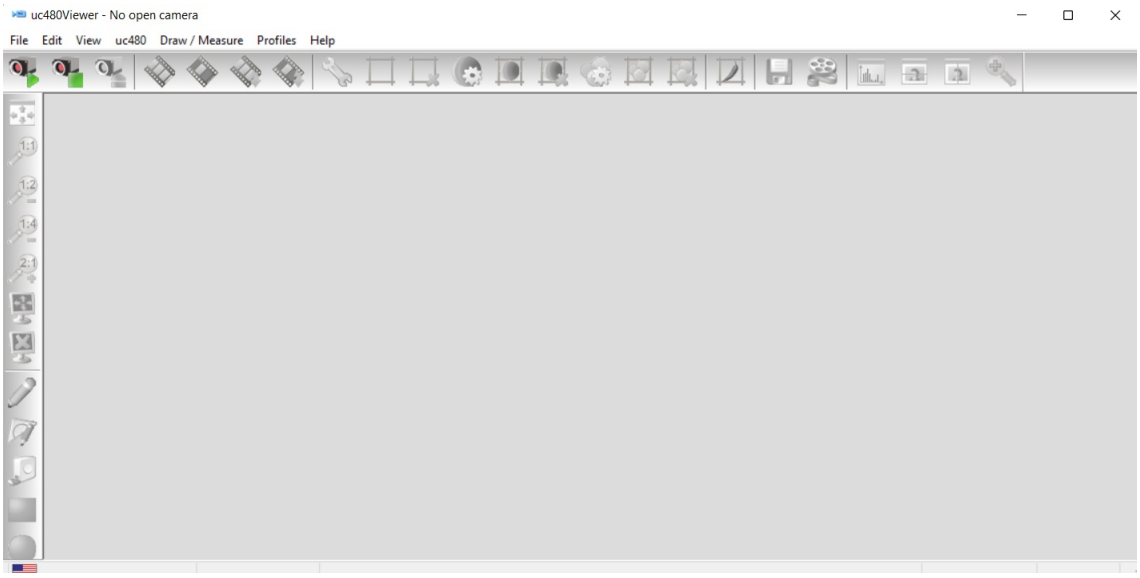

Figure 6: Open camera and start in the live mode.

5. Select the `Camera properties` option from the top toolbar and choose 1024×768 as the capture resolution. Also, click on `Auto contrast` and `Auto whitebalance` to adjust the exposure. You may need to click on the `Scale display to window size` on the left toolbar to fit the video into the viewing screen.

6. Choose the $40\times$ objective lens and use the focusing knob to bring the sample into focus. With a $40\times$ lens, you will need to move the sample upwards and as close to the lens as you can to start seeing some particles on the software. For more details on setting up the microscope and obtaining a sharp focused image, refer to the microscope's operation manual uploaded on the website.

7. During focusing, try not to hit the slide with the objective lens. This will pinch the cover slip and cause the trapped fluid to start a capillary movement which doesn't seize easily. Video analysis cannot be performed with a current of particles.

8. Once, some particles are in focus and vibrating in their places, proceed to the next step.

3.4 Video recording and calibration

1. Click the `Record video sequence` option on the top toolbar. A `Record dialog` window will appear on the screen. Click on the `create` option and select a video save path and start video recording.

2. After placing the slide under the microscope, the fluid may take a considerable amount of time to stem any currents of fluid taking place due to capillary action. Wait until the particles have stopped coursing through the slide and start to vibrate in their own places.

3. By clicking the `record` button, make a video of approximately 10s duration. (You'll have to stop the recording manually.)

4. Record as many samples as required and close the capture window afterwards.

5. Now replace the sample slide with `Motic` calibration slide. Using the same 40× lens, focus any one of the round dots of known dimension. To measure the diameter of the dot, first the camera must be calibrated to a scale by following these steps:

   (a) Go to `Draw/Measure` drop down menu at the top and select `Set measure unit` from `Measure` option. It must be set to 1.

   (b) Now draw a diameter line segment. The dimension line will show the length in pixels.

   (c) Divide the actual diameter length by the number of pixels measured. Enter this value in the `Set measure unit` field in the dialog box.

6. Note down the μm to pixel ratio as it will be used in the analysis.

3.5 Preparing Data for Quantitative Analysis

I hope the reader has already studied our "Primer on video tracking" uploaded on the website. To track the position of the microspheres, we will be using blob detection and tracking method for this experiment. This method is very efficient to detect and track small objects on clean backgrounds (similar to the videos we get from the microscope).

Download and extract on hard-drive the latest release of PhysTrack from our website. In Matlab, browse to the extracted PhysTrack directory where you will see the "+PhysTrack" folder in the "Current Folder" window. Run the script `analyzeBrownianMotion` inside the object tracking directory. The script will present, in series, some user-interfaces which will automate the whole process. First, an `open file dialogue` will appear. Browse and select your video file.

3.5.1 Cropping and Trimming

Each video frame may contain tens of small microspheres, simultaneous processing of which may fail because the objects tend to fade out during the video. This is why a crop region should be determined which contains up to 25 objects only. The

script will automatically show a tool to trim and crop the video. Use the `slider` and the `go-to buttons` to seek different frames of the video. Mark as `In-Frame` the moment you want your analysis to start and as `Out-Frame` the moment you want to stop tracking the microspheres. Use the `Reset Crop Region` button whenever you need to redefine the crop region. Preview the trimmed video if necessary and close the video trimming tool afterwards.

3.5.2 Binary Conversion

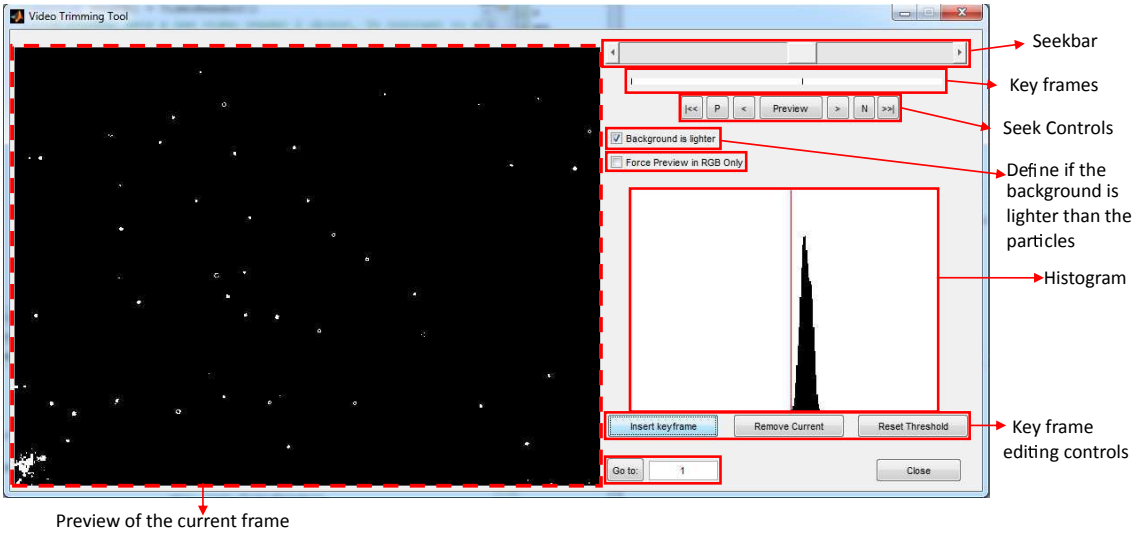

Figure 7: A screenshot of the binary conversion GUI showing its different parts.

The cropped image is then converted to grayscale and each pixel's lightness is compared with a threshold variable. The pixels lighter than the threshold have greater lightness value than the threshold and are considered as the background. All remaining pixels are considered to be the microspheres (the dark objects). Background pixels are converted to 0 (logic false to represent the background) and dark microsphere pixels are converted to 1 (logic true to represent the objects). It is clear that the determination of this threshold variable is very critical in the whole video analysis.

For this purpose, the script will now open a tool to convert the RGB video to binary. See Figure 7 which shows a screenshot of this GUI. Confirm that the `Background is lighter` checkbox is checked. To help determine the background threshold, a lightness histogram is shown on the left of the GUI. The histogram is an intensity-frequency plot of the grayscale image pixels. On the horizontal axes are the lightness levels from 0 to 255 and on the vertical axes is the count of the pixels sharing the corresponding lightness level.

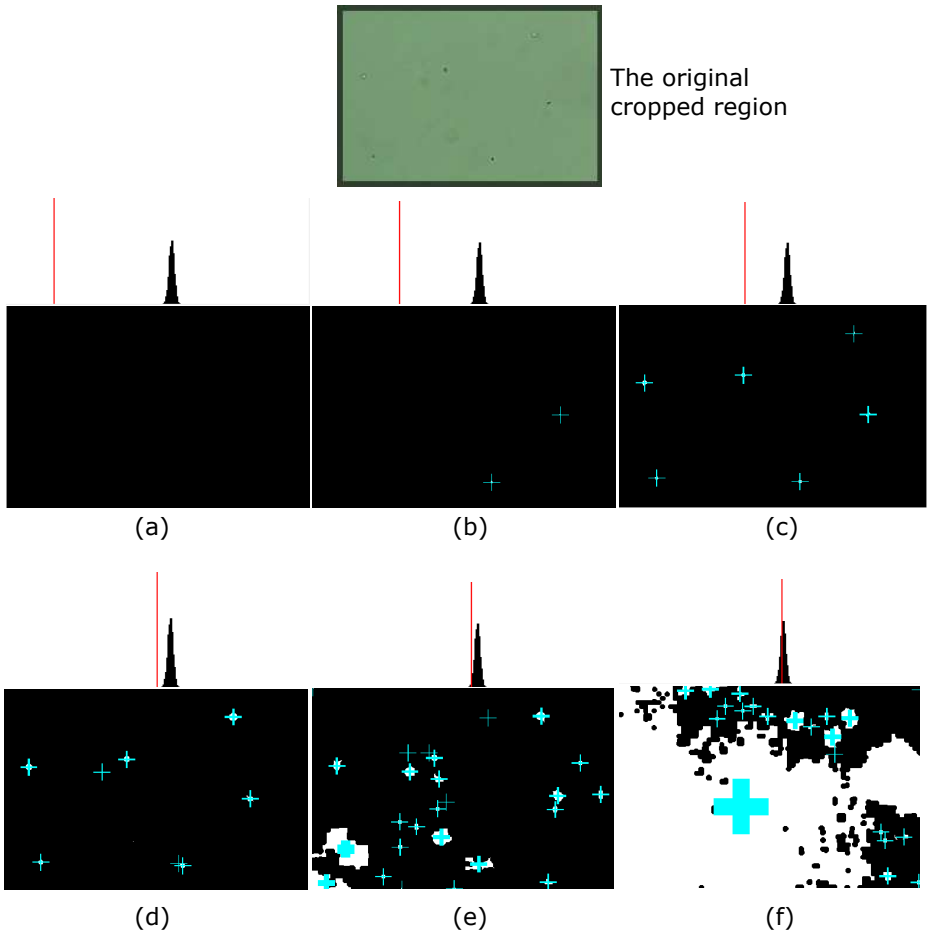

Figure 8: At the top, the original cropped region is shown. When the threshold is very small, no objects are detected (a). By increasing the threshold, the number of detected objects increase (see through (a), (b), (c) and (d)). A very good object detection can be seen just before the base of the central peek (see (c) and (d)). Threshold should be smaller than the base of the central peek, otherwise, most of the background also starts to pose as the required objects (see (e) and (f)

To understand the histogram, consider the pixels which construct the background of the microspheres. These pixels have colors very close to each other and, subsequently, share very similar lightness in the grayscale image. These pixels constitute more than 99 percent of the whole image and thus a peek can be seen in the middle of the histogram representing the background pixels. If we select a threshold which is slightly lesser than the base of this peek, we can clearly draw a line between the background and the microspheres.

A red vertical line is also displayed in the histogram which represents the threshold. To reset this level, press `Reset threshold` button and click anywhere in the histogram to see the results. To have a view of how varying the threshold level affects the detected objects, see Figure 8.

Once a threshold is set, preview the video using the play controls to confirm if the desired objects are detected throughout the video. Although this step is not necessary but if it is desired to have a continuously varying threshold with time, after resetting the threshold on a desired frame, click `Insert keyframe` button which will lock the threshold on that frame. Now, selecting some other frame and inserting another keyframe creates a continuously varying threshold which will automatically change between the inserted keyframes. As many keyframes can be added as required using this process.

3.5.3 Object Tracking

After closing the binary conversion tool, the script will allow to manually mark the objects to be tracked. You may select the objects manually or let the tool detect binary objects based on the threshold. Objects can be unselected manually by using `Remove objects` button and drawing a rectangle around the undesired objects. Close the tool when only the desired objects are selected and visible in the objects listbox. The script will start tracking the objects. A preview window will show the positions of the tracked objects with cross-hair mark. During the tracking, some objects may become invisible and the trajectory may be lost. A red cross-hair mark will be displayed to identify those objects. Once an object is lost, it is labeled as `invalid object`. The positional data of an invalid object is conserved but is not used in the analysis anymore.

It is important to know how the video tracking script works. Using threshold information for each frame and selected objects positions, the script precesses the video frames one by one. In every frame, each detected object's position is matched with the previous position to identify the index of each new object. This is how the script also knows which objects are lost with time and which appear amid the process.

3.5.4 Results Compilation

Once the mean square displacement of all particles is known some more variables are collected from the user using message boxes. These variables include the sample temperature, viscosity and average sphere diameter. At the end of a successful run, the script will leave some variables in the main workspace. See Table 1 to for the details of these variables.

${r_p}^2$ of a particle is its squared displacement from the initial position,

$$ {r_p}^2 = x^2 + y^2 \tag{12} $$

and ${r_n}^2$ is the mean squared displacement of all the valid particles,

$$ {r_n}^2 = \sum_{1}^{N} \left({x_i}^2 + {y_i}^2 \right) \tag{13} $$

Physical Quantity	Variable Names
Data of the particles	*particle.tp1, particles.tp2, ...*
Trajectories of the particle	*particle.tp1.xy, particles.tp2.xy, ...*
Square of displacements of the tracked particles	*particle.tp1.rp2, particles.tp2.rp2, ...*
Time stamps	t
Mean square displacements of all valid particles	*rn2*
$\mu\ m$ per pixel constant used	*ppum*

Table 1: Base workspace variables generated by the script.

3.6 Calculation of Boltzmann constant k_B and Diffusion constant D

Boltzmann's constant k_B is calculated using the slope of the $\langle r_n^2 \rangle$ vs t curve. The slope value after conversion of units of displacement from pixel^2 to μm^2 is substituted into Equation (9). Likewise, the diffusion constant is calculated using Equation (10).

See Table 2 to find out the value of dynamic viscosity of water at the required temperature.

Temperature (^oC)	**Viscosity** ($mPa.s$)	**Temperature** (^oC)	**Viscosity** ($mPa.s$)
2	1.6735	19	1.0266
3	1.619	20	1.0016
4	1.5673	21	0.9775
5	1.5182	22	0.9544
6	1.4715	23	0.9321
7	1.4271	24	0.9107
8	1.3847	25	0.89
9	1.3444	26	0.8701
10	1.3059	27	0.8509
11	1.2692	28	0.8324
12	1.234	29	0.8145
13	1.2005	30	0.7972
14	1.1683	31	0.7805
15	1.1375	32	0.7644
16	1.1081	33	0.7488
17	1.0798	34	0.7337
18	1.0526	35	0.7191

Table 2: Viscosity of water at different temperatures.

You can use the least-square curve fitting function `PhysTrack.lsqCFit()` to fit the processed data to a linear line. The syntax for using this function is

```
fitResult = PhysTrack.lsqCFit(t, rn2,...
            'rn2', 'm * t + c' , 't');
% fitResult.m will contain the slope of linear fit.
```

Q 5. Calculate k_B and D for your data. Refer to [6] to find the value of viscosity at the temperature of laboratory.

Q 6. Find the uncertainty in the value of k_B and D.

Q 7. Calculate Avogadro's number N_A using the value of slope and Equation (11). What is the uncertainty in N_A?

References

[1] Ronald Newburgh, Joseph Peidle and Wolfgang Rueckner, "Einstein, Perrin and the reality of atoms: 1905 revisited", Am. J. Phys. 74(6), 478-481 (2006).

[2] Stephen J. Blundell and Katherine M. Blundell, "Concepts in Thermal Physics", Oxford university press, (2008), pp. 207-208, 368-384.

[3] Dongdong Jia, Jonathan Hamilton, Lenu M. Zaman and Anura Goonewardene, "The time, size, viscosity, and temperature dependence of the Brownian motion of polystyrene microspheres", Am. J. Phys. 75(2), 111-115 (2007).

[4] http://www.pma.caltech.edu/ mcc/Ph127/b/Lecture16.pdf

[5] http://www.polysciences.com

[6] http://www.viscopedia.com/viscosity-tables/substances/water.