THIRD YEAR LABORATORY

LASER TWEEZERS

1 Aims

To investigate the phenomenon of laser tweezing.

To measure the viscosity of simple and complex fluids.

To quantify the trapping forces applied.

2 Objectives

1. To quantify Brownian motion of microscopic particles in solution.
2. To use Brownian motion to calculate the viscosity of the medium.
3. To perform force measurements on fat globules in water.
4. To measure the dependence of the spring constant versus the laser power for trapped particles.
5. To investigate surface effects on trapped particles.

3 Introduction

Optical tweezers use the fact that light is able to exert a force on matter. This is not intuitive, since we don't see this in everyday life because the forces are too small. We know that quantum theory predicts that light carries momentum,

$$\mathbf{p} = \hbar\mathbf{k} \tag{1}$$

and therefore

$$|\mathbf{p}| = \frac{h}{k} = \frac{E}{c}\,. \tag{2}$$

The prediction actually predates quantum mechanics and radiation pressure was suggested by Maxwell's e.m. theory (1873):

"*In a medium in which the waves are propagated there is a pressure in the direction normal to the wave and numerically equal to the energy contained in unit a volume.*"

Even earlier Kepler (1619) hypothesised the effect whilst trying to determine the nature of comet tails. Early experiments to observe and measure this radiation pressure were hindered by thermal (radiometric) effects. In 1901 (before the invention of the laser) two independent results were published by Lebedev and Nichols & Hull, who observed radiation pressure by looking at the effect of arc lamps on thin vanes. Lasers became available in the 70s, providing much higher intensities. Arthur Ashkin at Bell Labs pioneered optical manipulation in its current form and was awarded the Nobel prize for physics in 2018.

Ashkin's key insight was to use transparent particles in a transparent medium and he made use of Fresnel reflection from the particle interfaces. In his first paper (1970, [2]) he used a weakly focused laser beam (from a long focal length, low NA lens) and observed particle guiding; particles were drawn into the beam and travelled in the beam direction. The fact the particles were drawn into the beam led him to determine there was a gradient force caused by the intensity gradient across the beam.

A further key result was that by using two counter-propagating beams, particles could be trapped through a combination of radiation pressure (also now known as the scattering force) and the gradient force. The next year Ashkin did some work using a single beam and trapped a particle by directing the beam upwards and balancing radiation pressure against gravity [3]. He also used another beam coming in from the side to move the particle up or down within the trap. When the laser was turned off, he noted that the particle returned to the equilibrium position without oscillating, which indicated that the system was overdamped.

In 1986 Ashkin, Dziedic, Bjorkholm and Chu (another Nobel Laureate) demonstrated a new way to trap particles using a single laser beam [4]. They used a high numerical aperture microscope objective lens to trap a particle not only in the transverse direction, but also in the axial direction. In this way particles could be trapped with the beam pointing downwards against the force of gravity.

They demonstrated that they could trap particles in the range 25 nm to 10 μm and discussed how this could be used:

"*They also open a new size regime to optical trapping encompassing macromolecules, colloids, small aerosols, and possibly biological particles*".

In 1987 Ashkin published the first paper to trap viruses and bacteria. Later in 1987 he trapped cells using a near IR laser that did not damage the cells due to the reduced absorption.

It is recommended that you look up the above references and familiarise yourself with the background to optical tweezers e.g. read 'Optical Tweezers: Principles and Applications' by Philip Jones et al [1].

4 Theory

4.1 Photon momentum

We have seen the momentum of a photon is given by equ. (1) and therefore

$$|\mathbf{p}| = \frac{h}{\lambda_0} = \frac{E}{c} = \frac{h\nu}{c}\,, \tag{3}$$

where we have introduced the vacuum wavelength, λ_0.

In a medium of refractive index n, the wavelength reduces to λ_0/n, so the momentum of the photon increases in the medium. This is not what you might expect, and is still a hotly debated topic. It is known as the Abraham-Minkowski controversy, named after Abraham who claimed the momentum is hv/nc and Minkowski who claimed it is hnv/c. From equ. (3), Minkowski appears to be correct, and this is what we will use. The controversy revolves around the fact that we cannot really define the momentum of a photon in a medium, since it is questionable whether the photon model actual holds up.

If we take light travelling in a medium of refractive index n, incident on a perfectly reflecting mirror, then the magnitude of the change in momentum of each photon, $|\Delta\boldsymbol{p}|$ will be

$$|\Delta\boldsymbol{p}| = \frac{2nhv}{c} \quad . \tag{4}$$

For a light beam of power P, the number of photons per second, N, is

$$N = \frac{P}{hv} \quad , \tag{5}$$

so the magnitude of the force on the mirror, $|\boldsymbol{F}|$, will be

$$|\boldsymbol{F}| = \frac{2nhv}{c}\frac{P}{hv} = \frac{2nP}{c}. \tag{6}$$

This gives rise to what we know as *radiation pressure*. If the mirror is not perfectly reflecting, then to calculate the force on the mirror we simply multiply equ. 7 by the reflection coefficient.

For laser tweezers, where we are dealing with particles that are often spherical and transparent, we need to sum over all the photon trajectories (rays) through, and reflected off, the particle. This can be very complicated, and the calculated forces rarely match up with the measured forces. To overcome this issue a trap efficiency, Q, is defined so that

$$|\boldsymbol{F}| = \frac{QnP}{c}, \tag{7}$$

where n is the refractive index of the medium in which the particle is immersed.

4.2 Mie versus Rayleigh regimes

The size of the particle relative to the wavelength determines whether we are in the Mie regime or the Rayleigh regime. If

$$2r \leq \frac{\lambda}{\pi}, \tag{8}$$

where r is the particle radius, then we are in the Rayleigh regime and we have to treat the particle as a radiating dipole. In the opposite regime

$$2r \gg \frac{\lambda}{\pi}, \tag{9}$$

then we are in the Mie regime and we can use the ray optics approach as above.

4.2.1 Mie regime

If we take a transparent spherical particle, then we can model how the rays interact with the particle (Figure 1). At each interface we need to consider both a reflected and a transmitted wave. The reflection and transmission coefficients will depend on both the angle and the polarization, and can be calculated using the Fresnel equations. Due to the resultant forces from the momentum transfer, the particle will always want to move towards the focus of the laser beam. In terms of the actual forces involved, we can split them into two. One is the scattering force, $\boldsymbol{F}_{scat}$, which is due to back reflection from the particle, and the other is the gradient force due to the gradient of the electric field, $\boldsymbol{F}_{grad}$ (this acts in both axial and transverse directions). The total force is the sum of these two.

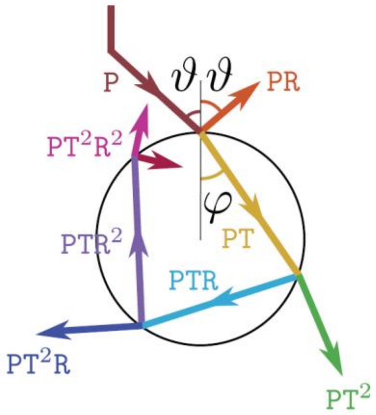

Figure 1: Ray model for a spherical particle. The incident beam of power P has reflection and transmission coefficients R and T at the interfaces respectively. θ is the angle the ray makes to the normal to the particle surface (J. Sanders PhD thesis, 2012).

4.2.2 Rayleigh regime

In the Rayleigh regime the particle is much smaller than the wavelength and the ray method cannot be used to calculate forces. We need to consider the induced dipole and the subsequent re-radiated light. The magnitude of the scattering force, $|\boldsymbol{F}_{scat}|$, is given by

$$|F_{scat}| = \frac{I_0 \sigma n_m}{c}, \qquad (10)$$

where σ is the scattering cross section, I_0 the incident light intensity and n_m the refractive index of the medium. The scattering cross-section is given by

$$\sigma = \frac{128\pi^5 r^6}{3\lambda^4}\left(\frac{m^2-1}{m^2+1}\right)^2 \qquad (11)$$

where r is the radius of the particle and m is the ratio of the refractive indices of the medium and particle,

$$m = \frac{n_p}{n_m}. \tag{12}$$

The strong dependence on wavelength due to the fourth power of λ in the denominator of equ. (11) explains why the sky is blue.

The magnitude of the gradient force, $|\boldsymbol{F}_{grad}|$, depends on the gradient of the intensity,

$$|\boldsymbol{F}_{grad}| = \frac{2\pi\alpha}{cn_m^2}\nabla I_0, \tag{13}$$

where the polarizability, α, of the particle is given by

$$\alpha = n_m^2 r^3 \left(\frac{m^2-1}{m^2+1}\right)^2 \quad . \tag{14}$$

Unfortunately most particles that are used for trapping fall between the Mie and Rayleigh regimes and therefore a more complete electromagnetic theory is required. However, equ. 7 can be used, because Q can accommodate any uncertainties in the theoretical model.

4.3 The trapping force

It is often very important to measure the trapping force, to quantify the quality of the trap (e.g. for the optical alignment) and when the trapped particle is used to perturb its environment (e.g. the force applied on a molecule).

To model the trap we treat it as a Hookean spring, with the particle as an over-damped oscillator. We assume that for a small distance away from the centre of the trap the trapping force is proportional to the displacement i.e. it acts like a spring.

Considering one dimension, $x(t)$, the Langevin equation of motion is

$$m\ddot{x}(t) + \gamma\dot{x}(t) + \kappa x(t) = F_B(t) \tag{15}$$

where m is the mass of the particle, γ is the viscous drag coefficient of the medium, κ is the optical trap stiffness and $F_B(t)$ is the force due to thermal fluctuations. The first term in equ. 15 is the inertial force, and can be neglected since most trapping is in the low Reynolds number regime, $Re << 1$, where viscous forces dominate.

4.3.1 Stoke's drag method

The viscous drag is given by Stoke's law

$$\gamma = 6\pi\eta r \tag{16}$$

where η is the dynamic fluid viscosity. Hence the Stoke's viscous force, $\boldsymbol{F_S}$, (the second term in equ. 15) is

$$F_S = -\kappa\dot{x}(t) = 6\pi\eta r\dot{x}(t) \tag{17}$$

If the force due to thermal fluctuations is ignored (they are small, but we will use them later on), then equ. 15 becomes

$$F_T = -\kappa x = F_S = \gamma \dot{x} \tag{18}$$

where F_T is the trapping force. Because the Stoke's force increases with velocity, one way to measure the trapping force is to trap a particle, then move the container in which it is trapped at ever increasing velocities until the Stoke's force exceeds the trapping force. At this point the particle will leave the trap and the velocity at which it does so will allow the Stoke's force (and hence the trapping force) to be calculated.

4.3.2 The equipartition method

Each degree of freedom in thermal equilibrium has $k_BT/2$ thermal energy. We can equate the thermal energy to that of a particle in a 1D optical trap (modelled as a harmonic potential), therefore

$$\frac{1}{2}k_BT = \frac{1}{2}\kappa\langle x^2\rangle \tag{19}$$

where $\langle x^2\rangle$ is the particle's positional variance. The probability, $P(x)$, of finding a particle at x within a potential $U(x)$ is described by a Boltzmann distribution as

$$P(x) \propto exp\left(-\frac{U(x)}{k_BT}\right) \quad . \tag{20}$$

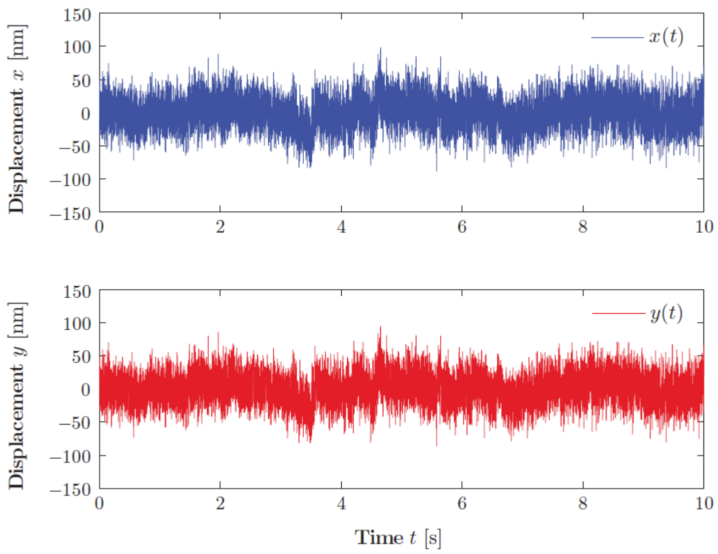

Figure 2: An example of positional vs time data for a trapped sphere. The fluctuations are due to thermal forces on the sphere.

$P(x)$ is proportional to $H(x)$ which is a histogram of particle positions. We take the harmonic form

$$U(x) = \frac{1}{2}\kappa(x - x_0)^2 \quad (21)$$

and therefore

$$H(x) \propto P(x) = Cexp\left(-\frac{\kappa}{2k_BT}(x - x_0)^2\right) \quad (22)$$

This is the form of a Gaussian.

Experimentally we measure the position as a function of time and then calculate *H(x)*. We then fit a Gaussian to determine $\langle x^2 \rangle$ and therefore κ. This does not require a knowledge of γ. See Figure 2 for an example of position data versus time for *x* and *y*. Figure 3 shows the corresponding histogram for this data with a Gaussian fit. Also shown is the calculated form of the potential for this data.

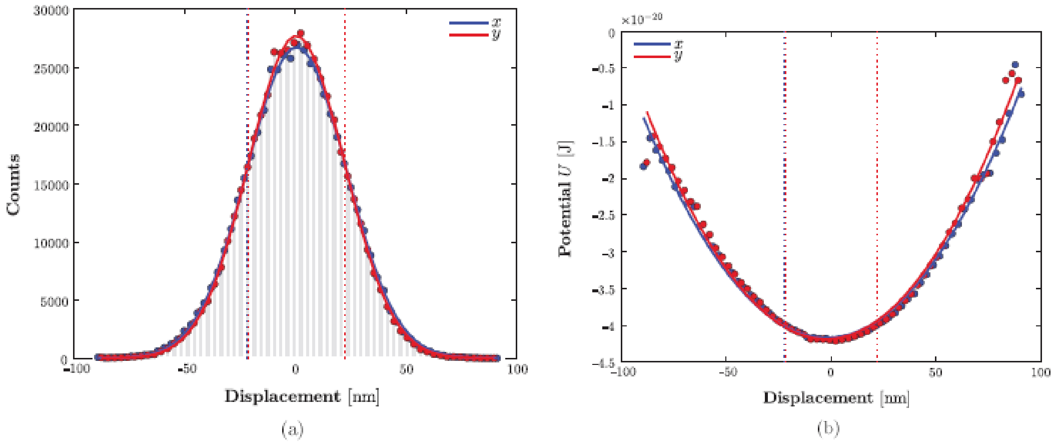

Figure 3: The same data in Figure 2 plotted as a histogram with a Gaussian fit (a), and the corresponding potential plot (b) as a function of displacement.

4.3.3 Autocorrelation function (ACF) and mean square displacement (MSD)

We can solve the over-damped Langevin equ. (15) for *x(t)* to give the autocorrelation function (ACF). This decays exponentially and is given by

$$\langle x(t+\tau)x(t)\rangle = \langle x^2 \rangle e^{-\tau/\tau_c} \quad (23)$$

where τ is a delay known as the autocorrelation time and τ_c is the decay time

$$\tau_c = \frac{\gamma}{\kappa} . \quad (24)$$

Substituting for γ and κ allows the viscosity to be calculated from

$$\eta = \frac{\tau_c k_B T}{6\pi r \langle x^2 \rangle} . \quad (25)$$

The Mean Square Displacement (MSD) of *x(t)* is given by

$$\langle \Delta x^2(\tau)\rangle = \langle (x(t+\tau) - x(t))^2 \rangle. \quad (26)$$

This is related to the ACF by

$$\langle \Delta x^2(\tau)\rangle = 2\langle x^2 \rangle - 2\langle x(t+\tau)x(t)\rangle \quad (27)$$

and therefore

$$\langle \Delta x^2(\tau)\rangle = \frac{2k_BT}{\kappa}\left[1 - exp\left(-\frac{\kappa\tau}{\gamma}\right)\right] \tag{28}$$

and so we can see that the ACF and MSD are complementary. See Figure 4 for example ACF and MSD plots.

Looking at equ. (23) and from Figure 4, when $\tau \to 0$, we just get $\langle x^2\rangle$, i.e. free diffusion, and in this region the plot is approximately linear. For $\tau \to \infty$, the $ACF \to 0$ and the $MSD \to 1$, and the particle is confined in the trap.

The MSD is easy to calculate from the data, and you will use this. Without the laser turned on (i.e. no trap), the MSD can also be used to investigate Brownian motion. With the laser on and a microsphere trapped, then plotting the MSD on a log plot gives a linear relationship, which can then be used to calculate τ_c for insertion into equ. (25).

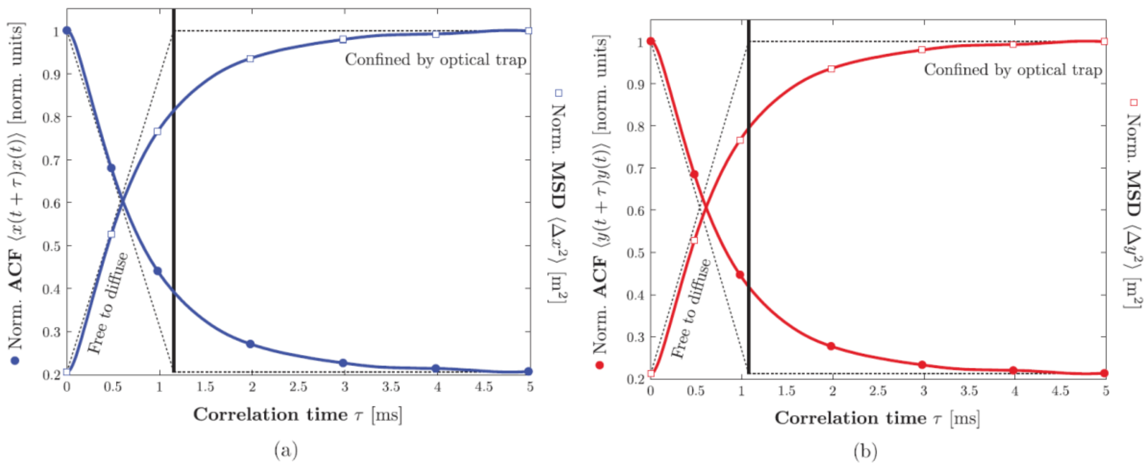

Figure 4: Autocorrelation function (ACF) and Mean Square Displacement (MSD) plots as a function of correlation time for optically trapped particles.

To generate the MSD data, the position (along one axis, for example the x-axis) of a microsphere can be tracked using the camera. Since the camera records a sequence of frames equally spaced in time, the time averaged MSD after N frames can be determined from

$$\langle \Delta x^2(t_n)\rangle = \frac{1}{N-n}\sum_{i=1}^{N-n}(x_{i+n} - x_i)^2 \tag{29}$$

where $n=1, \ldots, N-1$ and the expression averages over all the displacements of duration $n\Delta t$. Due to the number of samples in the averaging process the MSD should be reasonably smooth at short times and have much more significant noise at longer times.

5 Carrying out the experiment

5.1 Equipment

The basis of the equipment is a laser tweezer kit (EDU-OT1/M) from Thorlabs [5]. Their website has a detailed list of the components in this kit and there is a manual that includes details of possible experiments.

The basic laser tweezer kit comprises a collimated laser diode, which is passed through a high numerical aperture microscope objective lens to form a highly converging focussed optical trap. Before the objective lens, the beam passes through a collimating lens, two beam expanding lenses, reflects off two mirrors and then goes through a beam splitter. A white light LED is used to illuminate the sample, with an image formed on a CCD camera after the light passes through the beamsplitter. A small volume of liquid is contained in a well on a microscope slide. This sample can then be manipulated in all three axes using translation stages, two of which are moved using motors controlled from the computer, and the other via a micrometer (the axial direction). A schematic of the optical set up is shown in Figure 5.

There are a number of pieces of software you will need. The camera is run through ThorCam, and the laser diode and the motorised translation stages through Kinesis. The Thorlabs manual explains the basic functions. You MUST NOT exceed the maximum current through the laser diode, and that the translation stages cannot move negative. Consider setting the stages to around 1 mm before starting. Don’t use the HOME button. You should spend some time familiarising yourself with how to control the laser and translation stages. Good control of the sample is essential for this experiment.

You will also need to use ImageJ to convert the .tiff files to .avi, and Blender to track the particles. Appendices A and B explain how you can do this.

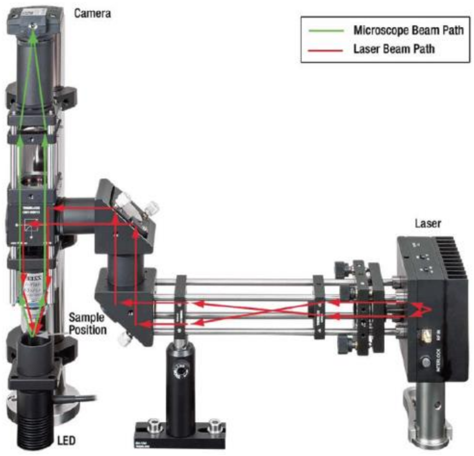

Figure 5: The optical paths for the laser and image formation (the sample stage is omitted for clarity) in the Thorlabs optical tweezers.

5.2 Samples

There are a number of different samples you can use for this experiment. The majority consist of small microspheres in solution. There are silica beads, polystyrene beads and melamine beads. You can also make up your own spheres of fat or oil in water using either cream or immersion oil. The density of the sphere (compared to the surrounding liquid), the size of the sphere and the refractive index of the sphere can all affect measurements.

To prepare a sample using the microscope slides with a well, a cover slip is needed that has been sealed in place with nail varnish. Be careful not to contaminate the concentrated source of microspheres. You will need a very small quantity of microspheres, diluted down greatly. Label your slides carefully so that you can identify them.

5.3 Experiments

There are a number of experiments you can carry out. Suggestions include:

1. Investigate Brownian motion of microspheres of different sizes by measuring the MSD.
2. Use equ. (25) to evaluate the viscosity of the medium.
3. Tweeze cream and/or oil particles and carry out force and/or spring constant measurements using both the equipartition theory and the escape velocity method. This can be done as a function of laser power.
4. Look at the effects of trapping a particle close to a surface.

5.4 Extensions

Successful extensions have included:

1. To trap particles in standard foods (gels, mayonnaise, starches etc.) and calculate their viscoelasticity.
2. To create an oil/water interface and push a trapped particle through the interface.
3. To use alternative types of tracking software. Blender is robust, but fairly inaccurate. Gaussian trackers such as Polyparticletracker show improved performance. Convolutional neural networks can also provide 3D tracking.

6 SAFETY

The laser used in this experiment is NOT eye safe! If the direct collimated beam were to enter your eye, damage to your retina WOULD occur. However, the tweezer system is quite safe as long as you DO NOT try to access the collimated beam. The collimated beam is contained within the cage system (contained within a box) and exits through the objective at which point it is highly diverging and safe. You MUST NOT remove either of the mirrors, or the

cover block on the beamsplitter, whilst the laser is on. Nor must you attempt to put reflective objects into the cage system which could accidentally or intentionally reflect out the collimated beam.
Immersion oil for the objective lens is poisonous if ingested. Take care to wash your hands after using it.

7 References

[1] P.Jones, O.Marago, G.Volpe, 'Optical Tweezers: principles and applications', CUP, 2015. The .pdf can be downloaded from the UoM library.

[2] A.Ashkin. Acceleration and trapping of particles by radiation pressure. *Physical Review Letters*, 24(4):156–159, 1970.

[3] A.Ashkin and J.M.Dziedzic. Optical levitation by radiation pressure. *Applied Physics Letters*, 19(8):283–285, 1971.

[4] A.Ashkin, J.M.Dziedzic, J.E.Bjorkholm, and S.Chu. Observation of a single-beam gradient force optical trap for dielectric particles. *Optics Letters*, 11(5):288, 1986.

[5] Thorlabs. Thorlabs tweezer kit. https://www.thorlabs.de/.

Previous authors: M.Dickinson, D.Wei.

Last updated: T.A.Waigh, October 2019

APPENDIX A

1. Blender environment

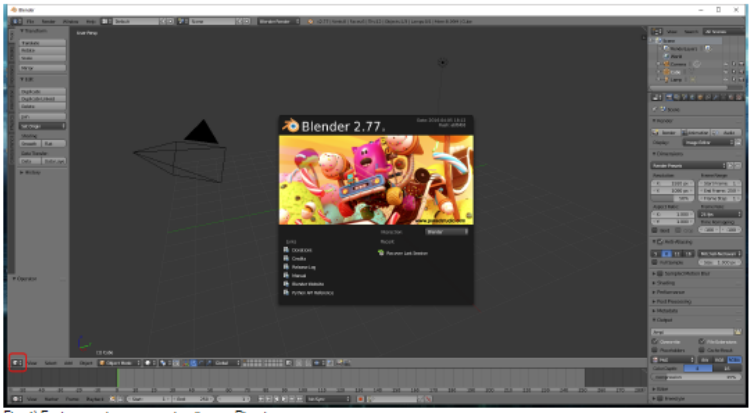

Fig. 1) Environment seen upon loading up Blender.

You will first need to change Blender to the 'Movie Clip Editor' mode.

- Access the editor menu by clicking the button highlighted in Figure 1.
- Select 'Movie Clip Editor' or press 'M'

2. Opening your video file in Blender

Click on the 'Open' button, found next to the editor menu. Browse to the correct directory and select your video file.

Fig. 2) Open file menu

The screen will now display the first frame of your video file, along with tracking settings (left), frame settings (bottom) and display settings (inside right). You need not worry about

rendering settings (outside right). The mouse scroll wheel can be used to zoom in/out on the video.

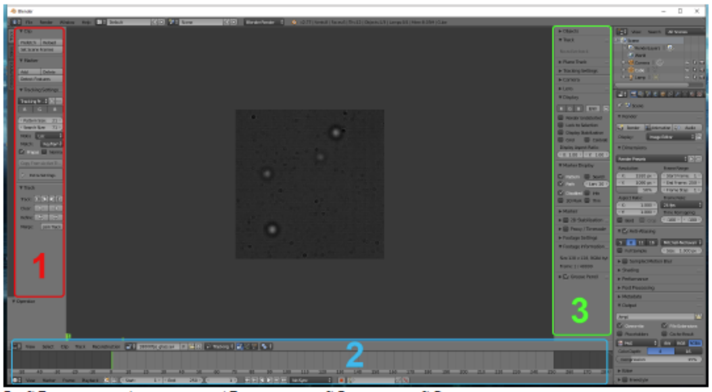

Fig. 3) The video editing/tracking screen. 1) Tracking settings. 2) Frame settings. 3) Display settings.

To greatly improve tracking speed, you should prefetch all the video frames into the computer's memory.

- Enter the number of frames to track into the frame settings area (Figure 3, section 2) and press 'Enter'.
- If you are unsure how many frames are in your video, you can view the amount by expanding 'Footage Information', found on the inside right menu (Figure 3, section 3).
- Now click 'Prefetch' in the upper left corner of the screen (Figure 3, section 1).
- The bar beneath the video preview will show the prefetching progress.

You can select your position in the video by clicking/dragging either the blue bar beneath the video preview, or the frame counter bar at the bottom of the screen. Alternatively, you can jump to a specific frame by entering it into the box to the right of the 'Start: | End:'. 'Shift + Left Arrow' or 'Shift + Right Arrow' are useful shortcuts to jump to the start or end frames, respectively.

3. **Marker settings**

Before adding tracking markers, ensure the following settings are (Figure 4).

Fig. 4) Tracking Settings

```
Motion    = Loc
Match     = Previous frame
Prepass   = True
Normalize = False
```

If, for some reason, the light intensity changes throughout your video, then better results may be found by ticking 'Normalize'.

'Pattern size' is the size of the feature you wish to track, in pixels. This should be set to be just larger than the size of the particles you are tracking.

'Search size' is the area the software will search for motion of the tracked object, between frames. Avoid having a search area which overlaps with a similar looking particle, as the track may jump onto the other particle.

Fig. 5) Marker Display

You may find it easier to adjust pattern/search size manually once markers have been added to the video. Ensure 'Pattern' and 'Search' are ticked in the 'Marker Display' menu on the right of the screen (Figure 5). The size of the marker can then be adjusted by dragging the handles on the screen.

4. **Tracking**

You can add a tracking marker to the particle you wish to track by either:

- Clicking 'Add' in the top left, then clicking on the desired location on the video
- Ctrl + Left-Click the desired location on the video

The tracking marker should now be visible on the screen, as seen in Figure 6a. You can click and drag the marker to move its position. By clicking and dragging the track preview window (Figure 6b) you can fine-tune the position of the marker.

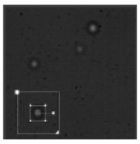

Fig 6a (Left): Video preview including tracking marker.

Fig 6b (Right): Track preview can be used to fine-adjust the position of the tracking marker.

You can either track markers individually or multiple markers at the same time.

Fig. 7) Start tracking by pressing the highlighted button, left of the screen.

- Ensure the current frame is where you wish to start tracking. (To go to the start of the video, press 'Shift + Left Arrow'.
- With your marker(s) selected, press the '>' button on the left of the screen (Figure 7)
- Tracking should now be performed, with the path of the particle shown on the screen.
- If tracking stopped before the end of the video, you might want to either adjust your tracking settings or add and track a new marker from where it finished.
- Markers can be merged by selecting them and clicking the 'Join Track' button.

5. **Exporting tracking data**

Once you are finished tracking, you will need to export your data to a CSV file.

- Click the editor menu button (bottom left) and select 'Text Editor' or press 'T'.
- Now click 'Open' and browse to the provided Python script.
- Once open, you will want to change the export directory by editing the line `'filepath = "/my_folder/"'`
- By default this will save to `C:\my_folder\`
- Now press 'Run Script' at the bottom of the screen.

You will now find an individual CSV file for each of your tracked particles. The files will be named by video file name and track number. The CSV file will have three columns.

1) Frame (number)
2) X-Position (pixels)
3) Y-Position (pixels)

APPENDIX B

Converting .tif stack to .avi

By default, ThorCam saves video files as a .tif stack (multipage TIFF file). Most tracking software requires video input to be in the form of an .avi file.

ImageJ (or Fiji) provides the most convenient way to perform this conversion.

1. Load ImageJ
2. File > Open then browse to the desired .tif file
3. Allow all frames to load, shown by a progress bar
4. File > Save As > AVI…
5. Set compression as 'None' to ensure no loss of quality
6. It does not matter what 'Frame Rate' is set to
7. Click 'OK' and choose a filename for your new .avi file

APPENDIX C

Python script for exporting tracking data

This file might be on your computer called blender_export.py

```
# Blender tracking export to CSV
# Python script originally taken from:
# http://scummos.blogspot.cz/2012/11/blender-exporting-camera-tracking.html
# https://gist.github.com/anonymous/31c915d611f0a84e5d33
# Last edited by David Wei, University of Manchester, August 2016
import bpy
import csv
D = bpy.data

frameNums = True  # include frame numbers in the csv file
relativeCoords = False  # marker coords will be relative to the dimensions of the clip
filepath = "/my_folder/"
markers = {}


for clip in D.movieclips:
    print('Clip {} found'.format(clip.name))
    if relativeCoords:
        width = 1
        height = 1
    else:
        width = clip.size[0]
        height = clip.size[1]
    for ob in clip.tracking.objects:
        print('Object {} found'.format(ob.name))
        for track in ob.tracks:
            fn = '{}_{}_tr_{}'.format(clip.name.split('.')[0], ob.name, track.name)
            markers[fn] = []
            print('track {} found'.format(track.name))
            for framenum in range(clip.frame_duration):
                markerAtFrame = track.markers.find_frame(framenum)
                if markerAtFrame:
                    coords = list(markerAtFrame.co.xy)
                    coords = [coords[0] * width, coords[1] * height]
                    markers[fn].append(coords)


for key, value in markers.items():
    print(key)
    filename = "{}{}{}".format(filepath, key, ".csv")
    open_data = open(filename, 'w', newline='')
    with open_data as writer:
        writer = csv.writer(writer, delimiter=',', quotechar='"', quoting=csv.QUOTE_NONNUMERIC)
        if frameNums:
```

```
    # writer.writerow([key, "x", "y"])
    for i, data in enumerate(value):
        writer.writerow([i] + data)
else:
    # writer.writerow(["x", "y"])
    for data in value:
        writer.writerow(data)
```