CORRELATING SUPER-RESOLUTION WITH TRANSMISSION ELECTRON MICROSCOPY TO CHARACTERIZE NANOPARTICLES

Cristina Izquierdo Lozano

TREBALL FINAL DE GRAU BIOTECNOLOGIA
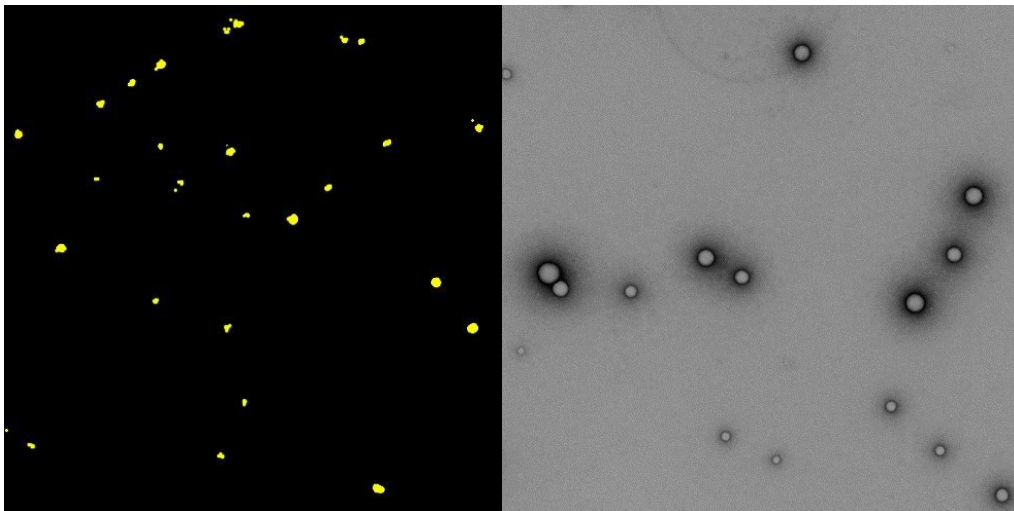
Academic tutor: Maria Montserrat Poblet Icart, Biotecnologia, Departament de Bioquímica i Biotecnologia, montserrat.poblet@urv.cat

Cooperating with: Institute for Bioengineering of Catalonia, IBEC

Supervisors: Teodora Andrian, PhD fellow, Nanoscopy for nanomedicine, tandrian@ibecbarcelona.eu, and Sílvia Pujals, Senior researcher, Nanoscopy for nanomedicine, spujals@ibecbarcelona.eu

January 2020

Index

1. Information on the centre

The Institute for Bioengineering of Catalonia (IBEC) is a research centre whose purpose is to carry out interdisciplinary research at the highest international quality level which, by creating knowledge, helps to improve health and quality of life and generate wealth.

IBEC focuses on the interaction between scientific disciplines and technologies which are apparently separate from one another in order to find revolutionary solutions in health and life sciences.

IBEC's goal is to generate new technological advances of key importance to innovation which can be applied to many different areas like early diagnosis, new therapies based on regenerative medicine, better quality of life compatible with an ageing population, and technological advances to increase efficiency and make healthcare sustainable.

1.1. Nanoscopy for nanomedicine

The Nanoscopy for Nanomedicine group at IBEC uses Super-Resolution Microscopy (nanoscopy) to visualize and track in living cells and tissues self-assembled nanomaterials with therapeutic potential (nanomedicine).

The understanding of materials-cell interactions is the key towards the development of novel nanotechnology-based therapies for treatment of cancer and infectious diseases.

Thus, the group aims to use a multidisciplinary approach to develop novel nanomaterials for the treatment of cancer and infectious diseases.

The aim of the group is the development of novel nanocarriers for drug delivery based on self-assembly. In particular, the group focuses on the development of self-assembled nanoparticles (NP) and nanofibers able to selectively target diseased cells and deliver locally therapeutic moieties such as drugs and genetic material.

Super-resolution microscopy is used to track nanomaterials during their voyage in the biological environment and to visualize the interactions with blood components, immune system and target cells. This is achieved thanks to a variety of super-resolution techniques based on single molecule detection such a stochastic optical reconstruction microscopy (STORM), photoactivated localization microscopy (PALM), point accumulation for imaging in nanoscale topography (PAINT), and single particle tracking (SPT). Super-resolution microscopy provides a molecular picture of structure-activity

relations and represent a guide towards the design of innovative materials for nanomedicine.

2. Abstract

Nanoparticles (NP) used for the delivery of therapeutics have significant advantages over conventional chemotherapeutics as they can encapsulate both hydrophilic and hydrophobic molecules and provide physical protection to the encapsulated drug. Additionally, they present enhanced delivery of the drugs and tumour specific accumulation thanks to the enhanced permeation and retention (EPR) effect. However, the lack of standardization in the characterization of the NP is a major impediment for their approval for clinical applications.

For this reason, three different techniques have been used in this study for the characterisation of poly(lactic-co-glycolic acid)-polyethylene glycol (PLGA-PEG NP), in order to standardize experimental protocols for the development of novel nanomedicine-based therapeutics. These techniques consist of Dynamic Light Scattering (DLS), Stochastic Optical Reconstruction Microscopy (STORM) and Transmission Electron Microscopy (TEM). More specifically, the two main objectives were a) to analyse and compare the size uniformity of NP formulated either manually or using a microfluidic device, and b) to quantify the encapsulation efficiency of these methods by using STORM.

DLS showed a 75% increase in the average size of the NPs when compared to STORM and TEM. The width in the half of the height for the different formulations was 20 nm in the case of the formulations with the microfluidic device and of 35 nm for the manual formulations.

Thus, NPs were more homogeneous and better controlled when formulated using a microfluidic device, in terms of size and size distribution. Furthermore, STORM has been proven a useful method for NP characterisation and, more importantly, it has proven to be a promising and more precise method to quantify the encapsulation efficiency at a single particle level.

CLEM: *Correlative Light and Electron Microscopy* | STORM: *Stochastic Optical Reconstruction Microscopy* | TEM: *Transmission Electron Microscopy* | SRM: *Super-resolution Microscopy* | SMLM: *Single-Molecule Localization Microscopy* | NP: Nanoparticle

3. Introduction

3.1. Nanoparticles

3.1.1. Nanoparticles for Drug Delivery in Medicine

The concept of nanomedicine has been introduced in the last century and since then many revolutionary developments have been made in this area. Nanomedicine, the medical application of nanotechnology, is a discipline which is applied to various fields including pharmaceuticals, imaging agents and theranostics. With a focus on nanoparticles (NP) used for delivery of therapeutics, 50 formulations have successfully translated into the clinic, with liposomes being the first nanodrug delivery system to achieve this (Ventola, 2017). The first PEGylated liposome-based product was Doxil, the liposomal formulation of doxorubicin for the treatment of both haematological and solid tumours. Conventional Doxorubicin treatment presents cardiotoxicity, nausea and vomiting and myelosuppression, however these side effects are reduced when the drug is encapsulated in the liposomal formulation. Apart from liposomes, other formulations have been approved by the FDA including polymeric, nanocrystal, micelles and inorganic NP (Abraham et al., 2005; Bulbake et al., 2017).

NP have some advantages in medical applications over conventional chemotherapeutics. Some can encapsulate both hydrophilic and hydrophobic molecules, and by doing so they provide physical protection against hydrolysis and oxidation of the encapsulated drug. Additionally, they present enhanced delivery and retention of the therapeutic to the site of action through a process called passive targeting. Tumour specific accumulation is one of the main advantages of nanocarriers, and this is due to the enhanced permeation and retention (EPR) effect. In tumours, the abnormally wide pores in the blood vessels allow for the extravasation of materials with sizes up to several hundreds of nanometers. This, together with the absence of lymphatic drainage, leads to a relatively effective and selective accumulation of nanomedicines in tumours (Golombek et al., 2018). Furthermore, by designing nanocarriers with specific targeting moieties, these can deliver and release the encapsulated drugs directly at the site of action, through a process called active targeting. Overall, this leads to a reduction of collateral effects caused by the drug's toxicity and a reduction of the required dose to be administered as well as enhanced specificity of the treatment.

Thanks to many of their features such as small size, improved solubility, and multi-functionality, new multifunctional devices that can target, diagnose, and treat diseases

such as diabetes or cancer can be designed and developed (Bhatia, 2016; Singh et al., 2009; Khan et al., 2019).

NP-based systems are one of the novel parenteral formulations that could guarantee drug stability, both in storage and in physiological conditions, and which allow the release of bioactive molecules in a sustained and controlled way. Drugs administered via parenteral routes have many advantages over drugs taken orally, such as improved absorption, high bioavailability of drugs (Gulati et al., 2011) and a very quick onset of action (Date et al., 2008). The parenteral route also provides a targeted and sustained release of drug in predictable manner (Gulati et al., 2011).

3.1.2. Challenges of Nanoparticles in Nanomedicine

Despite the great research hype in the field of nanomedicine in the past century, there are many impediments preventing a large majority of NP formulations from reaching the clinic. A significant challenge in NP development and approval is their characterisation with respect to safety and toxicity. In contrast to small-molecular drugs, NP vary greatly with respect to formulations and applications, hence each novel formulation must be thoroughly characterised. Furthermore, the variability of published literature regarding the characterizations performed and experimental details reported on nanoengineered materials constitutes a significant barrier to progress; therefore, a "minimum information standard" needs to be created for experimental literature. Material and biological characterization and details of experimental protocol have to be given in order to improve reproducibility, increase quantitative comparisons of bio-nano materials, and facilitate meta analyses and *in silico* modelling (Faria et al., 2018).

A major impediment to characterizing nanomaterials is a lack of standardization. This lack of standardization and difficult regulation turns into less approval of the NP for clinical applications. To solve this, the International Standards Organization has an active technical committee (ISO/TC 229) developing new standards for nanotechnology (Faria et al., 2018).

3.1.3. Polymeric Nanoparticles

Although there are various types of nanoparticles approved and under investigation, polymeric NP are one of the most widely used in nanomedicine. This is mainly due to their ease of synthesis and proven safety and efficacy (Ventola, 2017). The NP used in this project are formulated using two different polymers: poly(lactic-co-glycolic acid) (PLGA) and polyethylene glycol (PEG).

PLGA is a biodegradable, biocompatible, and relatively hydrophobic copolymer that is synthesized by the co-polymerization of two different monomers, the cyclic dimers (1,4-dioxane-2,5-diones) of glycolic acid and lactic acid.

PLGA undergoes hydrolysis in the body to produce the original monomers: lactic acid and glycolic acid. These two monomers under normal physiological conditions are products of various metabolic pathways in the body, thus they are endogenous and easily metabolized by the body via the Krebs cycle. For this reason, PLGA has a minimal systemic toxicity in biomedical applications and is one of the most successfully developed biodegradable polymers (Danhier et al., 2012; Kumari et al., 2010).

PEG is a hydrophilic and non-ionic polymer used to coat NP. The main advantage is that it repels opsonin proteins due to steric repulsion, and thus avoid the opsonisation process of the NP. This results in an increase in the blood circulation half-life of the formulation by several orders of magnitude (Owens III et al., 2006). Moreover, PEG has low toxicity, so is commonly used in medical and biological applications.

3.2.   Standard methods for size analysis

3.2.1. Dynamic Light Scattering

Dynamic Light Scattering (DLS) is a common technique for measuring the size of particles in the sub-micron region, suspended within a liquid. The random movement of particles (Brownian motion) occurs due to the bombardment by the solvent molecules that surround them. A laser beam is passed through the sample, and analysis of the fluctuations of the scattered light is used to calculate the hydrodynamic diameter of the NP. Small particles cause the intensity to fluctuate more rapidly than the large ones, so the larger the particle, the slower the Brownian motion is. The velocity of the Brownian motion equals to the Translational diffusion coefficient (D) (Malvern Instruments, accessed August 2019; Stetefeld et al., 2016).

The value that refers to how a particle diffuses within a fluid is the hydrodynamic diameter, which is the diameter of a sphere that has the same translational diffusion coefficient as the particle.

The primary result from DLS is the mean value from the intensity distribution (called the z-average) and the polydispersity index (PDI) to describe the distribution width. In a homogeneous sample the PDI should not be higher than 0,200, a higher PDI would mean

the sample is too heterogeneous, that is, the difference between the smallest nanoparticle and the biggest is high (Stetefeld et al., 2016).

There are three different distributions obtained in the DLS results: intensity, number and volume. The primary results from DLS systems are typically reported as an intensity distribution, which is the most closely related to the raw signal, the number and the volume distribution are derived from this. The number distribution is the most comparable to image analysis methods or counting based techniques such as STORM and TEM (Stetefeld et al., 2016).

When comparing DLS data to microscopy techniques such as TEM, the number distribution would be most comparable since this represents the number of molecules in each bin in a histogram (Stetefeld et al., 2016).

3.2.2. Transmission Electron Microscopy

Transmission electron microscopy (TEM) is a very powerful tool for material science that has been widely applied to characterize material morphology. In TEM, a beam of electrons passes through a thin (<~100 nm thick) sample and achieves up to sub-nanometer resolution. (Hauser et al., 2017; Cohen, 1992; Winey et al., 2014; Tang et al., 2017). Generally, it is incompatible with wet samples due to its high-vacuum operating conditions. (Hauser et al., 2017). In order to enhance the sample contrast, negative staining with Uranyl Acetate 2% or UranyLess is a necessary step during sample preparation. The spatial resolution for TEM, that is the minimum resolvable distance, is about 1,5 nm (Grogger et al., 2003), although this is greatly experiment-dependent.

3.3. Super-Resolution Microscopy to Overcome Challenges in Nanomedicine

3.3.1. Optical Microscopy

Optical microscopy, also known as light microscopy, is the traditional form of microscopy. This technique is employed to closely view a sample through the magnification of converging lenses, placed between the sample and the eye, with only visible light (W. Davidson et al., 2002).

The power of magnification depends on the ocular and the objective lenses, for example, for a 10x ocular lens and 100x objective lens used together, the final magnification is 1000x (W. Davidson et al., 2002). However, the best spatial resolution which can be achieved with classical light microscopy is approximately 200-300nm lateral and 500-

800nm axial (Sanderson et al., 2014). This resolution is limited by the Abbe diffraction limit, which is proportional to the light wavelength and inversely proportional to double the numerical aperture of the objective (Monticone et al., 2014).

3.3.2. Fluorescence Microscopy

Fluorescence microscopy is a major tool in monitoring cell physiology. It makes use of fluorophores to stain tissue components, bacteria, and other pathogens. These fluorophores have made it possible to identify cells and sub-microscopic cellular components with a high degree of specificity. By using multiple fluorescence labelling, different probes can simultaneously identify several target molecules. The spatial resolution depends on the wavelength of the excitation light and the microscope optics, approximately 200nm lateral and 500nm axial (MicroscopyU, accessed September 2019).

The modern fluorescence microscope combines high performance optical components with computerized control of the instrument and digital image acquisition, leading to the imaging of single molecules and a wide range of sophisticated spectroscopic applications (MicroscopyU, accessed September 2019; Sanderson et al., 2014).

3.3.3. Single Molecule Localisation Microscopy

Super-resolution microscopy (SRM) bypasses the Abbe diffraction limit of optical resolution, allowing the visualization in detail of subcellular organization with the usual advantages of optical microscopy: sample preservation, imaging flexibility and target specificity (Pawley, 2006; Schermelleh et al., 2019).

SRM methods are based on wide-field (WF), total internal reflection fluorescence (TIRF) or confocal microscopy, differing in how the samples are excited and how the emitted photons are detected (Schermelleh et al., 2019; Hauser et al., 2017). These methods are generally divided into two categories: one overcomes the diffraction limit by engineering the illumination patterns and the other locates single molecules that undergo stochastic "on−off" fluorescence photo-switching (Hauser et al., 2017). The first includes Stimulated emission depletion (STED) and Structured Illumination Microscopy (SIM), while the second category encloses single-molecule localization microscopy (SMLM) methods, on which we will focus. SMLM methods include Stochastic optical reconstruction microscopy (STORM), points accumulation for imaging in nanoscale topography (DNA-PAINT) and photoactivated localization microscopy (PALM).

SMLM achieves sub-diffraction resolution, based on wide-field illumination, through single molecule switching by stochastic excitation and detection of fluorescent point emitters. As shown in Figure 1, small subsets of individual emitters are randomly activated or switched on/off in consecutive acquisitions. These signals are collected over several thousands of frames and later computationally processed to detect single molecules and determine their centre positions (Schermelleh et al., 2019). These are finally assembled through superimposition into a single-plane binary image (Schermelleh et al., 2019; Sauer et al., 2017). SMLM approaches can be implemented at low cost on conventional wide-field setups and most implementations can separate individual dyes between 20 nm lateral and 50 nm axial resolution (Hauser et al., 2017).
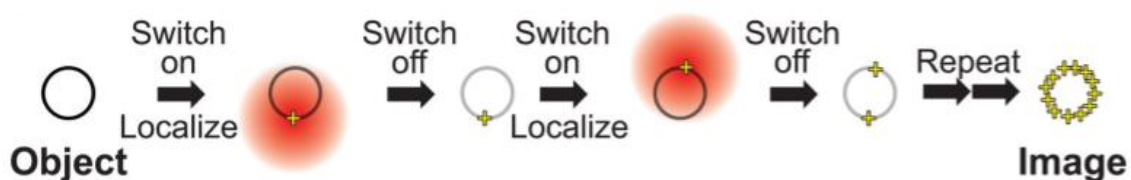
Figure 1. Principle of single-molecule-localization-based super-resolution imaging (Dempsey et al., 2011).

.

Current SMLM approaches differ primarily in how the switching is achieved: it can either use photoactivation, photo-switching of dyes or binding and fluorescent activation of specific dyes (Schoen et al., 2011; Schermelleh et al., 2019; Szczurek et al., 2017). Also, in the case of DNA-PAINT, transient oligonucleotide hybridization is used (Ehmann et al., 2014; Jungmann et al., 2016).

In these methods, the dye emission is photo-controlled, and the fluorophores can be switched on and off with laser illumination. Figure 2 shows the Jablonski diagram, a graphical explanation of what happens when a fluorophores in the ground state (off) is excited to a higher state, it returns to the initial state by emitting fluorescence. This is carried out to have a large majority of the dyes in the off state, and only a few sparsely distributed dyes are emitted. Because of this, the fluorescence of single markers can be detected, and the precise spatial positioning of the emitting molecules can be identified by fitting a Gaussian profile and obtaining the Gaussian centroid. Iterating this, a few tens of molecules are identified for every frame and the super-resolution image, up to 20-25 nm, is obtained by summing up the positions of all molecules detected (Pujals et al., 2019).

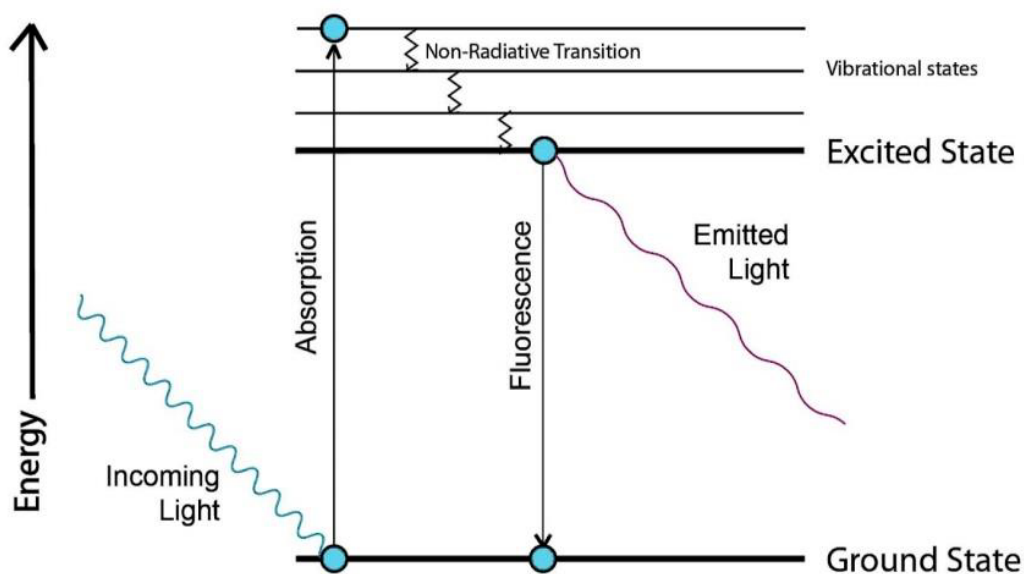*Figure 2. Jablonski diagram explaining a molecule excitation and return to the ground state, emitting fluorescence.*

3.3.3.1. Stochastic Optical Reconstruction Microscopy

Stochastic optical reconstruction microscopy (STORM), is currently one of the most employed methods to study nanoparticle internalization and trafficking (Pujals et al., 2019), as it enables the positioning and measurement of internalized particles down to 80 nm (Zwaag et al., 2016; Pujals et al., 2019).

The main advantages of STORM are very high resolution (around 20 nm), single molecule detection, simple microscope setup and quantification based on the localisations (Schermelleh et al., 2019). The last one provides the ability to get not only a precise measure of the size of a NP but also to quantify the amount of dye or drug encapsulated in NPs.

One of the main limitations of STORM is the necessity of adding cofactors to the solvent to achieve the desired photo-switching (Pujals et al., 2019).

Buffering the sample for STORM is important to help maintain the pH stability, since most fluorophores are very pH sensitive and their performance could be compromised at non-optimal pH. In addition, STORM buffer formulations usually include an oxygen-scavenging system (GLOX) since molecular oxygen is implicated in photobleaching. GLOX includes glucose, the enzyme glucose oxidase (an oxido-reductase extracted from the fungus *Aspergillus niger* that catalyzes the oxidation of glucose by molecular oxygen, yielding gluconolactone and hydrogen peroxide) and catalase, which

decomposes the hydrogen peroxide to prevent its toxicity in cells. Apart from the GLOX system, the STORM buffer also contains a reducing agent such as mercaptoethylamine (MEA), which intercalates in the fluorophores and prevents it from blinking so only ~1% of the fluorophores are active in each frame (Allen et al., 2019).

3.3.4. Correlative Light and Electron Microscopy

Correlative light and electron microscopy (CLEM), is the integration of two different microscopy techniques (light and electron microscopy), performed on the same sample, to produce results that emphasize the strengths of each technique while offsetting their individual weaknesses. (Hauser et al., 2017). In this project, the focus was on correlating STORM and TEM.

One of the main challenges in CLEM is the sample preparation; new strategies that are compatible with both the light microscope and the electron microscope need to be developed, since there are some incompatibilities between the sample preparation of each technique and the images obtained may be difficult to correlate. (Hauser et al., 2017).

The complementary advantages of light microscopy for TEM are that methods like STORM enable the observation of specific targets with excellent molecular specificity, contrast, sensitivity and multicolour fluorescence enables simultaneous visualization of multiple targets to probe interactions. (Hauser et al., 2017).

However, some of the difficulties found in CLEM approaches and methodology make this technique highly challenging. For example, steps for EM sample preparation may quench fluorescence, which could lead to strong background autofluorescence (Watanabe et al., 2011; Clancy et al., 1998; Hauser et al., 2017), and may alter the photo-switching capability and other photophysical parameters of the fluorophore critical to SRM. (Ha et al., 2012; Hauser et al., 2017). In order to perform CLEM, it is necessary to avoid any sample processing steps that are detrimental to fluorescence prior to SRM imaging and apply them after SRM is completed. First SRM is performed and then TEM, since the sample needs to be dehydrated for TEM but not for SRM (Hauser et al., 2017).

The alignment of the EM and SRM could be highly challenging and, during data collection, it may already be difficult to locate the same region. To aid in this matter, grids are used to help locate the same area, while low-magnification zoomed-out light microscopic images can help provide an overview of the context of the region of interest.

Fiducial markers which are both fluorescent and EM-visible can also be used in addition to the grids, for ease of correlation between the two images (Hauser et al., 2017).

4. Hypothesis

The first hypothesis of this research is that STORM can be implemented as a precise method for the physiochemical characterisation of NPs, specifically for the analysis of size homogeneity and distribution of different NP formulations.

The second hypothesis is that STORM can be implemented as a new technique to quantify the encapsulation efficiency of fluorescent molecules in NPs, and that it is more precise in doing so than using the standard approach of spectrophotometry. Moreover, STORM provides information at the single particle level.

4.1. Objectives

The first objective of this project is to combine and compare three different techniques (DLS, STORM and TEM) to analyse the size homogeneity of PLGA-PEG NP formulated via two different methods, namely manual nanoprecipitation and via an automated microfluidic device.

The second objective is to quantify the encapsulation efficiency of the fluorescent dye DiD (1, 1' – Dioctadecyl - 3, 3, 3', 3' - Tetramethylindodicarbocyanine, 4 - Chlorobenzenesulfonate Salt) in PLGA-PEG NP using STORM, and to compare the results achieved by analysis of low-resolution imaging intensity (spectrophotometry).

5. Materials and Methods

Materials used for this project were, for the NPs formulation, polymers were supplied by PolySciTech: PLGA-PEG (AK102, Mw PEG:PLGA 5:30 kDa, L:G in PLGA 50:50. Lot: 50331SMS) and PLGA (AP082, L:G 50:50, acid endcap, Mw. 25–35 kDa. Lot: 70201AMS-A).The dye DiI (1,1'-Dioctadecyl-3,3,3',3'-Tetramethylindocarbocyanine perchlorate, 4236) was provided by Sigma Aldrich and DiD (1,1'-Dioctadecyl-3,3,3',3'-Tetramethylindodicarbocyanine perchlorate , 4-Chlorobenzenesulfonate Salt) by Vybrant.

The equipment used for the characterization of the NPs are: DLS Malvern Zetasizer Nano – ZS, JEOL 1010 Microscope and Baltec CTA 005 Glow Discharge Unit for the characterization by TEM, and the Nikon N-STORM.

For TEM imaging, carbon-coated copper grids (200 mesh) were used, provided by Electron Microscopy Sciences, and 100nm gold NPs stabilized suspension in citrate buffer (molecular weight: 196,97 g/mol) used as fiducial markers were provided by Sigma-Aldrich (742031-25ML).

Finally, for the STORM imaging: microscopy slides (ref. BPB016, dimensions 76x26 mm, thickness 1/1,2 mm) were supplied by RS France, coverslip (ref. D102424, dimensions 24x24 mm, thickness 0,13-0,16 mm) were supplied by deltalab,and as fiducial markers Fluorescent Microspheres (TetraSpeck) by ThermoFisher scientific (100 nm diameter) were used.

5.1.   Nanoparticle Formulation

The formulation of the NP has been achieved by the PhD fellow, Adrianna Glinkowska.

Block copolymers PLGA-PEG and PLGA were used and mixed together. In particular, the chosen ratio for all the formulations was 85% PLGA-PEG + 15% PLGA.

A batch of 10 µM DiI loaded particles was prepared for the size distribution analysis (both via the manual and the microfluidic device methods).

For the encapsulation efficiency analysis (different concentrations of dye), batches of 1 µM, 10 µM, 50 µM and 100 µM DiD loaded NP were prepared via the manual nanoprecipitation method.

5.1.1. Manual formulation of PLGA-PEG Nanoparticles

The manual formulation is achieved via the nanoprecipitation method (Yadav et al., 2010). In brief, a system is mounted in a glass vial containing anti-solvent phase (acetonitrile) and the polymers PLGA-PEG and PLGA. The solvent phase (water) is injected drop by drop with a pipette into the vial, which is later maintained under magnetic stirring for about 5-6 hours to allow the evaporation of the anti-solvent phase.

5.1.2. Microfluidic Device formulation of PLGA-PEG Nanoparticles

The microfluidic system, a polydimethylsiloxane (PDMS) chip, is provided by two lateral inlet channels (intended for the anti-solvent injection), a central inlet for solvent phase (water) and an outlet for the particles collection.

5.2. Sample preparation for DLS

The translational diffusion coefficient is used to calculate the hydrodynamic diameter, as long as the other conditions are kept constant (i.e. the absolute temperature and viscosity of medium in which nanoparticles are suspended). It is therefore important to specify these constant values before beginning a DLS measurement. The following parameters were used: material: PLGA-PEG (refractive index: 1.460 and absorption: 0.0), dispersant: water (viscosity: 0.887 cP, refractive index: 1.33), temperature: 25°C, equilibrium time: 30 seconds, cuvettes: Quartz cuvettes ZEN 2112.

Before the use of the DLS equipment, NPs are vortexed for 1 minute and the cuvette is cleaned with filtered MilliQ water first and with acetone second. Next, the parameters described before are set in the DLS software and the equipment is calibrated using an empty cuvette. Finally, the analysis is run three time for each sample plus the control (filtered MilliQ water).To obtain the size of the NP, the DLS equipment follows Equation 1 (Stetefeld et al., 2016):

$$d(H) = \frac{kT}{3\pi\eta D}$$

*Equation 1. Stokes-Einstein equation. Where **d(H):** hydrodynamic diameter, **D:** translational diffusion coefficient, **k:** Boltzmann's constant, **T:** absolute temperature and **η:** viscosity.*

5.3. Sample preparation for TEM

The NP formulation is first vortexed for 1 minute prior to use and is then diluted x20 in MilliQ water according to its original concentration (10 mg/ml).

Carbon-coated copper grids are put under the UV glow discharge for 30 seconds. This makes the grids hydrophilic, so the nanoparticles can attach better.

As seen in Figure 3, a drop of approximately 40µl of NP formulation is placed on a parafilm, and the grid is set on top of the drop, (with the carbon film side) for 10 minutes to allow NP attachment. The same process is followed for the fiducial markers, but instead the grid is placed at the bottom and the drop on top.

The grid is then washed three times by placing it on top of filtered MilliQ water drops (1 minute, 30 seconds and 30 seconds).

Finally, the grid is placed on top of a filtered Uranyl Acetate (UA) 2% (in MiliQ water).drop during 1 minute for the negative staining. This stains the background, leaving the NPs untouched, so they can be seen under the microscope.
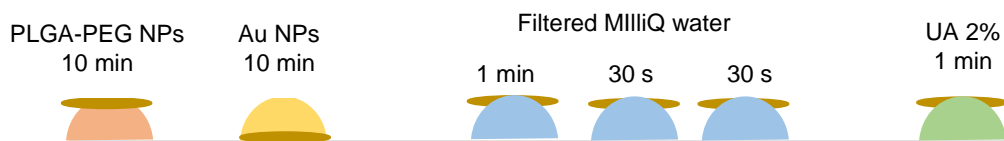
Figure 3. Sample preparation for TEM. Attachment of PLGA-PEG NP (orange), gold nanoparticles (Au NP, yellow), washing with MiliQ water (blue) and negative staining with UA 2% (green), with the individual timings specified. The copper carbon TEM grid is seen in brown.

5.4. Sample preparation for STORM

Microscope chambers are prepared by attaching a microscopy slide to a coverslip using double-sided cello tape as seen in Figure 4.
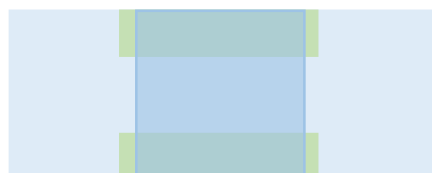
Figure 4. Microscope chamber. **Light blue:** *microscopy slide.* **Dark blue:** *coverslip.* **Green:** *double-sided cello tape.*

NP and TetraSpeck fiducials are independently vortexed for 1 min then diluted in PBS. The GLOX buffer is prepared with 80% of PBS, 10% of glucose 50% (w/v), 1% of glucose

oxidase and catalase (glox) and 10% of MEA 100 mM (reducing agent; 2-Mercaptoethylamine•HCl), in this order.

Approximately 40 µL of the sample is inserted in the chamber using a 200 µL pipette and left for 10 min, upside down, for proper attachment of NP to the coverslip. Then, a clean tissue is placed on the other side of the chamber to absorb the NP solution not attached, whilst a x1000 dilution in PBS of 100 nm TetraSpeck fiducial markers is introduced via the other side of the chamber and left to attach the same as the NP.

After the attachment of the NP and TetraSpeck fiducials, the chambers are washed twice with 40 µL PBS and then 40 µL of GLOX buffer is introduced. The chambers then are sealed with nail polish on the sides in order to keep the GLOX buffer from evaporating.

For the imaging, two laser-lines were used: fiducials were imaged once every 100 frames in 488 nm (~80 mW) at 5% of laser power, and NPs were imaged during 12k in 561 nm (~80 mW) for the DiI encapsulated NPs or 647 nm (~160 mW) for the DiD encapsulated NPs at 100% of laser power.

5.5.   Size Distribution Analysis of Nanoparticles

5.5.1. TEM analysis

After obtaining the images from TEM, size analysis in ImageJ is performed. First, the scale has to be set by drawing a straight line on the scale bar shown in the image and setting the known distance (Analyze > Set scale), in nanometers.

In order to perform an automated analysis, the image has to be pre-processed (Process > FFT > Bandpass Filter and set "Filter large structures" down to ~20 pixels, or the convenient value for each sample) and then select a threshold to get the lighter pixels, which are the inside of the nanoparticles (Image > Adjust > Threshold, set "Dark background" and "Over/Under", move the upper bar to zero and the bottom bar to the right position where the full inside of the nanoparticle is filled with green). After applying the threshold, the adjustments need to be set back to original values, so the NPs are seen in white and the background in black.

Once the image is processed and the threshold is set, the next step is to analyse the size of the nanoparticles by calculating the area of the white and circular clusters of pixels (Analyze > Analyze particles, select "outlines" in *Show* dropdown menu, check "Display results" and modify the minimum size and circularity according to the particles in the image). If the output is not correct, the parameters can be modified as many times as

necessary until most of the NPs are considered, as well as the use of the "Eraser tool" to get rid of some background noise which could be confused with a nanoparticle.

Finally, all the measurements are displayed on a list that can be exported or copy-pasted to an Excel file, for later calculations. The result displayed is the area of the nanoparticles, thus, to get the diameter, Equation 2 needs to be applied:

$$\text{Diameter} = \left(\sqrt{\frac{Area}{\pi}}\right) \times 2$$

Equation 2. To obtain the diameter of a circle from its area.

5.5.2. STORM analysis

As previously mentioned, the NIS elements NIKON software generates a list of all the individual localisations by Gaussian fitting of each blinking fluorophore and summarises the results in a high-resolution image, and a text file. A density filter is then applied to remove any unspecific blinking found on the glass coverslip, by setting the filter to a minimum of 20 localisations and a radius of 100 nm. Using a previously defined Matlab clustering algorithm (see annexes section 10.2), the txt. file containing the specific localisations is analysed to give a size distribution and quantification of number of dye localisations encapsulated per nanoparticle. First, the list of localisations in the dye channel is analysed (DiD = 647 nm, DiI = 561 nm) and a mean-shift clustering algorithm is applied to spot the clusters of localisations which correspond to specific nanoparticles and to cancel out false localisations.

The parameters used for the mean-shift clustering algorithm were: Bandwidth (cluster size in nm) = 150, MinPts (minimum number of localisations in a cluster) = 20, MaxDiam (maximum NP diameter in nm) = 150 and the optional parameter 'MinClustDist' (minimum distance between clusters in nm) was set to 80.

6. Results and discussion

6.1. Size distribution analysis

6.1.1. Size distribution: comparison between different formulation methods

The size and size distribution of PLGA-PEG NP (DiI encapsulated) was analysed and compared using three different techniques: DLS, TEM and STORM. The four types of PLGA-PEG NP formulations analysed were formulated via two different methods: a microfluidic device (A and B) and manual nanoprecipitation (C and D). By using different flows or ratios of solvent and anti-solvent B and C were intentionally formulated to have a smaller diameter (~60 nm), whilst A and D were formulated to have a larger diameter (~100 nm). The aim of this was to a) analyse and compare the three techniques in their ability to characterise physicochemical properties of nanomaterials, and b) to compare the size and size heterogeneity of nanoparticles formulated manually or via an automated microfluidic device. For specific formulation and analysis details please see section 5.1. The hypotheses are: 1) STORM can be integrated as a standard characterising tool in nanomedicine research, 2) the three techniques will provide different results on the size and size homogeneity of the formulations, due to their specific methodological requirements and resolutions, 3) that the microfluidic device formulation method will offer a more robust and homogeneous size distribution compared to the manual. Furthermore, the microfluidic device will be able to formulate NP with different diameters by altering the flow rate, whereas by the manual method NP are only formulated in one size.

The size and size distribution of the four formulations were first analysed using a standard technique, DLS, frequently used in the characterisation of NP formulations. As mentioned previously, DLS measures the hydrodynamic diameter and polydispersity index (PDI) of the whole sample in solution. As seen in Figure 5, there is no clear difference observed in the size distribution (width distribution) or PDI between the different formulations. A PDI of less than 0,200 suggests a homogeneous sample, indicating that all four formulations have a homogeneous distribution. However, the size difference between the smaller and larger sized formulations is evident in the case of the microfluidic device formulations (A mean size=81 nm and B mean size=50 nm) but not in the manual formulation (C mean size=69 nm and D mean size=59 nm). This is due to the augmented precision and control of the diffusion of materials when formulating NP using a microfluidic chip.
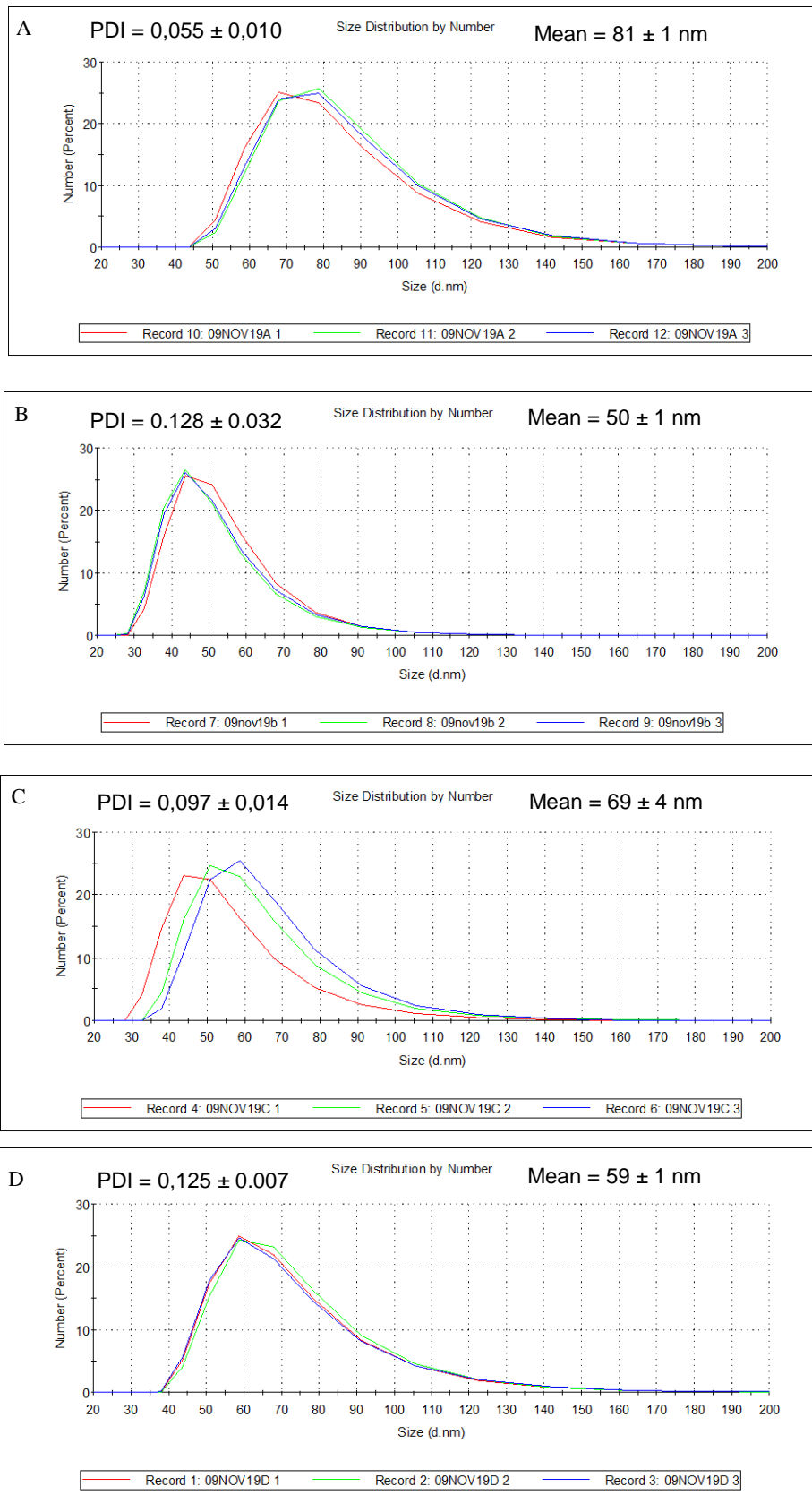
Figure 5. DLS size distribution by number for PLGA-PEG NPs formulated with a microfluidic device (A and B) and by manual nanoprecipitation (C and D). Mean size and PDI have been included in the figure, 3 repeats for each sample.

Next, the size and size distribution of the four formulations were analysed using the advanced microscopy technique TEM. Using this technique, the diameter of each nanoparticle imaged can be measured directly on a single particle basis, rather than the whole sample as in DLS.

Figure 6 shows the size distribution analysis by TEM. In the NPs formulated using a microfluidic device there is a clear difference in the distribution width at the half height between sample A (40 nm) and B (15 nm) but is not as clear between the samples formulated manually (C=20 nm and D=30 nm). Furthermore, as seen in the analysis by DLS, this formulation method shows an evident difference in the average size, the larger being 60 nm (A) and the smaller 36 nm (B). A and D were intentionally formulated with larger diameter, but again, this size difference is not seen clearly in the formulation by manual nanoprecipitation (C=51 nm and D=45 nm). These results can be correlated to the results shown in DLS and to the expected results, showing that the formulation by manual precipitation is difficult to control.
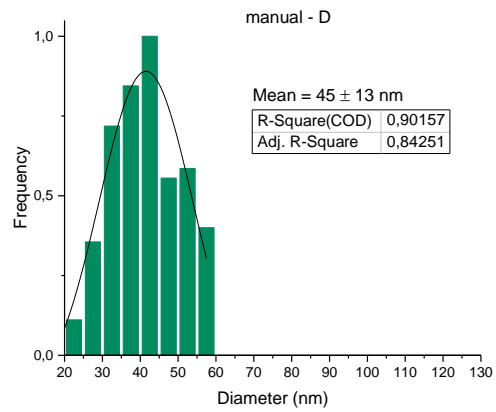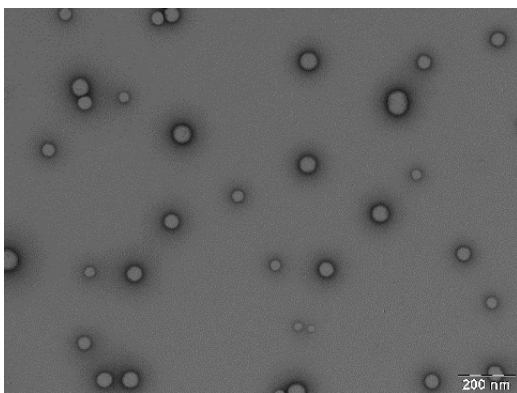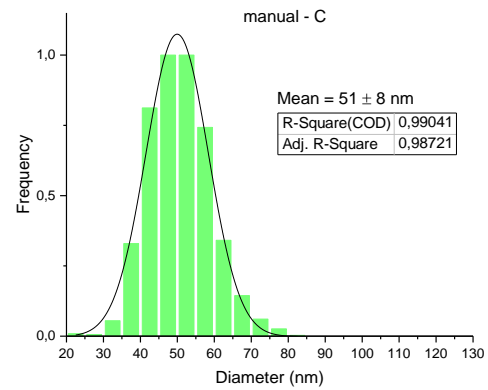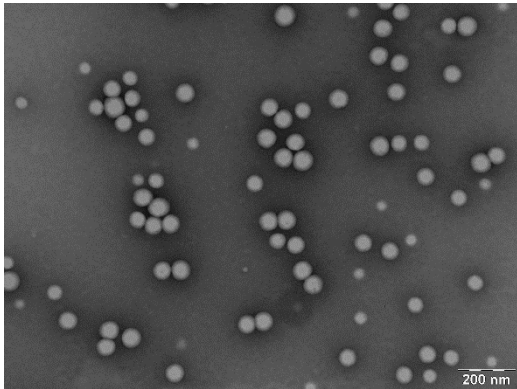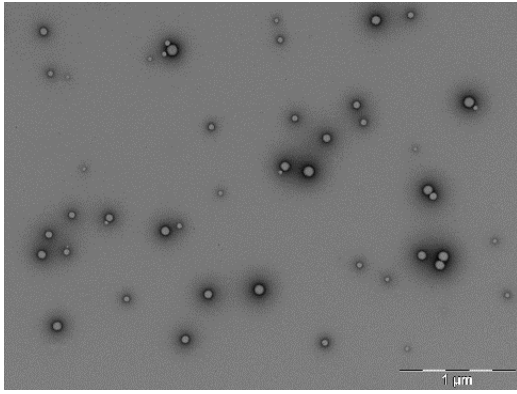
Figure 6. TEM size distribution analysis for PLGA-PEG NPs, formulated with a microfluidic device (A and B) and formulated manually (C and D). On the left panel example particles obtained with TEM, on the right panel TEM-based size distribution histograms. Mode and mean values are shown for each graph as well as the r-square and adjusted r-square values for the fitting in Gaussian distribution. Negative staining was achieved using uranyl acetate 2%. The number of NPs analysed in each formulation was between 200-1000 NP. The results were then normalised.

Finally, the size and size distribution of the four formulations were analysed using the super-resolution microscopy technique STORM. Using this technique, the diameter of each nanoparticle imaged can be measured, based on the localisations of the DiI dye (excitation=549 nm, emission=564 nm) encapsulated in the NPs. Using an automated Matlab image analysis script, individual NP are identified, and their size distribution is analysed through a size histogram.

Figure 7 shows the size distribution analysis by STORM. A subtle difference is observed in the distribution width between the two methods of formulating NPs, manual nanoprecipitation and using a microfluidic device. The samples C and D, formulated manually, show a wider distribution than the samples formulated with a microfluidic device. Even though when looking in the width in half of the height (15 nm in A, 20 nm in B and 30 nm in C and 25 nm in D) the difference is the same, the samples formulated manually show higher frequency of NPs of bigger size. Once again, this agrees with the hypothesis and the size distributions analysed by TEM. However, the difference in size average analysed by STORM is not as evident as it was with the other analysis methods. The manual nanoprecipitation formulation shows no difference in size (C=D=70 nm) while the formulation with a microfluidic device shows only a small difference (A=61 nm and B=57 nm). All the distributions show a good fitting (0,94 < r-square < 0,97).
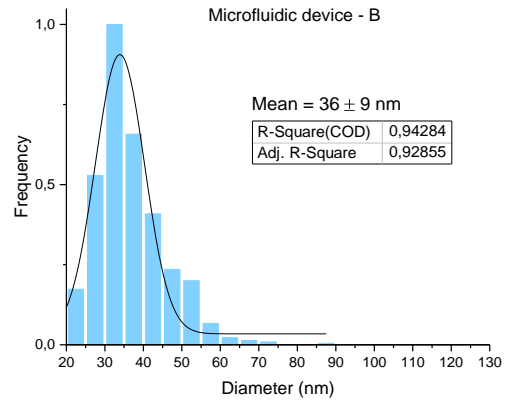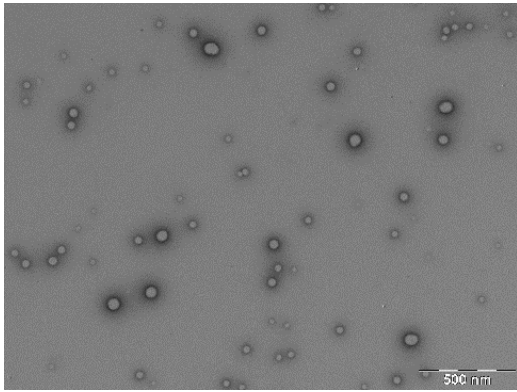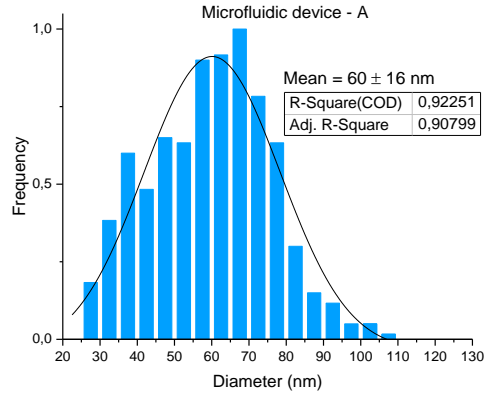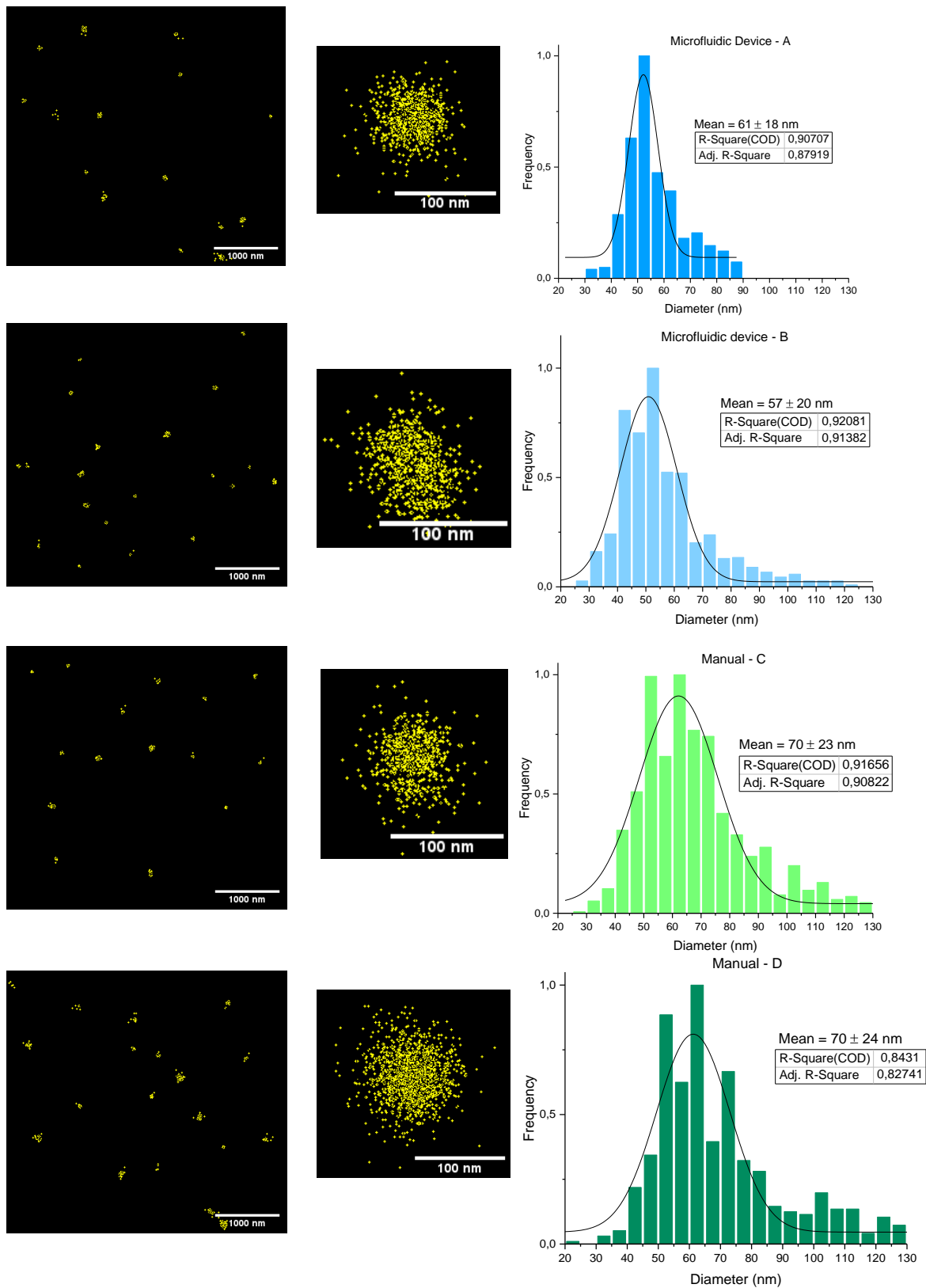
Figure 7. STORM size distribution analysis for DiI PLGA-PEG NPs, formulated with a microfluidic device (A and B) and formulated manually (C and D). **Left column**: *super-resolution image of the dye localisations of the nanoparticles.* **Center column:** *NP zoom-in super-resolution image.* **Right column**: *STORM based size distribution . Mode and mean values are shown for each graph as well as the r-square and adjusted r-square values for the fitting in Gaussian distribution. The number of NPs analysed in each formulation was between 200-1000 NP. The results were then normalised.*

Table 1. Size distribution analysis comparison with different analysis methods, DLS, STORM and TEM of different NPs formulation methods: microfluidic device (A and B) and manual nanoprecipitation (C and D).

Formulation	DLS		STORM		TEM	
	PDI	Mean (nm)	Half-height width (nm)	Mean (nm)	Half-height width (nm)	Mean (nm)
Microfluidic device (A)	0,055	81	20	61	40	60
Microfluidic device (B)	0,128	50	25	57	15	36
Manual nanoprecipitation (C)	0,097	69	35	70	20	51
Manual nanoprecipitacion (D)	0,125	59	35	70	30	45

Table 1 shows the differences in values between the different formulations (manual nanoprecipitation and microfluidic device) analysed with three different methods, DLS, STORM and TEM. When comparing DLS and STORM, there is no clear difference in the size average, but the size average between TEM and the other two methods (excluding the value for sample A) is clearly a 75% less than in DLS, and approximately the same for STORM.

The size difference between the three methods is due to the specific spatial resolution of each microscope and the different methodology involved in sample preparation. DLS is a light scattering technique able to calculate the NP size from the hydrodynamic diameter (Figure 8), which is defined as the size of a hypothetical hard sphere that diffuses in the same fashion as that of the particle being measured (Maguire et al., 2018). Since NPs in solution are non-spherical, dynamic and solvated, the diameter calculated is only indicative of the apparent size of the dynamic hydrated NP (Maguire et al., 2018). However, since

Figure 8. Difference between the hydrodynamic diameter (DLS) and the diameter measured by TEM (Maguire et al., 2018).

most applications of nanomaterials involve solutions, the hydrodynamic diameter is essential in correlating NP sizes with physiological responses (Maguire et al., 2018).

In contrast to light microscopy, TEM uses an electron beam instead of light. Since electrons have a wavelength smaller than that of light, TEM can achieve a much higher resolution than that from a light microscope. Specifically, it can differentiate nanoparticles separated by approximately 1,5 nm (Grogger et al., 2003). Furthermore, TEM (unlike STORM and DLS) provides direct structural information, which makes it a very powerful and accurate tool in characterisation of nanomaterials. However, unlike the other two methods and as seen in Figure 8, the sample in TEM is observed in a dry state, as the imaging is done under vacuum. The implication of this on PLGA-PEG NP is that the PEG polymer on the surface of the nanoparticles shrinks, due to the lack of counterions found

in solution, which generally repel the PEG chains from one another. This makes the NP appear smaller than in DLS and STORM.

Single molecule localisation microscopy techniques such as STORM have excellent spatial resolution (can differentiate between two NP when they are separated by approximately 20 nm) (Pujals et al., 2019) and can offer molecular specificity and single molecule sensitivity. However, in comparison to TEM, NP smaller than 20 nm are not accurately detected. Furthermore, the diameter analysed with STORM depends greatly on the photo-switching ability of the dye encapsulated, as the on and off blinking events are used to build a gaussian fit of the individual dye localisations. The size distribution therefore depends on the STORM buffer used, the physical properties of the encapsulated dye and the ability of the STORM buffer to reach the encapsulated dye.

Methods based on microscopy techniques like STORM and TEM provide information relating to the physical dimension of the NP, at the single particle level, whereas DLS provides information on the whole sample of NPs when it is in solution (Maguire et al., 2018). Additionally, as stated before, STORM measures the diameter based on the localisations of the dye encapsulated in each NP.

To sum up, each of these techniques come with their advantages and disadvantages, they are useful in particular ways. For this reason, the use of these different techniques, including STORM as a new technique in NP characterisation, gives a robust measurement that will improve the chances of NPs formulations to reach clinical applications.

6.2.   Encapsulation efficiency

6.2.1. Comparison b etween different dye concentrations

NP have been deemed a popular formulation strategy for improving therapeutic drug delivery and reducing side effects of chemotherapeutics as outlined in section 3.1, Remarkably however, on average only 0,7%, of the administered dose is actually delivered to a solid tumour (Wilhelm et al., 2016). This has negative consequences for the use of NPs in clinical applications as they have less effect on the tumour. For this reason, evaluating the encapsulation efficiency of NPs is vital to nanomedicine research in order to determine the amount of drug that is encapsulated and eventually released at the site of action.

The aim of this section is to use the super-resolution technique STORM as a novel approach to quantify the amount of dye encapsulated in PLGA-PEG NP. In STORM, the image reconstruction is based on single molecule localisation which allows the visualisation and quantification of fluorescent dyes encapsulated in the NP. For this research's purpose, the far-red fluorescent carbocyanine dye DiD (Excitation=644 nm, emission=665 nm) is used instead of a therapeutic drug for STORM analysis. As it is a lipophilic dye, it can be easily encapsulated in the PLGA core of the NP. Furthermore, it has good blinking behaviour, allowing imaging by STORM. It is expected that the lower concentrations of DiD show less localisations per NP.

Figure 9 shows a comparison of the encapsulation efficiency between formulations with different concentration of DiD dye: 1 µM, 10 µM, 50 µM and 100 µM, formulated via manual nanoprecipitation. The number of localisations per NP was obtained using the Matlab clustering algorithm in the annex section 10.2, then the results were normalised and plotted in a histogram and an average of all the numbers of localisations was done for each sample The higher the DiD concentration is the more localisations per NP are seen in STORM (1 µM=137 ± 200 loc./NP, 10 µM=458 ± 521 loc./NP, 50 µM=1340 ± 1157 loc./NP and 100 µM=1450 ± 729 loc./NP). The number of localisations is proportional to the number of molecules of dye, so these results are as expected. For 50 µM (Figure 9c) a second population starts to appear towards ~2350 loc./NP, which is more evident for 100 µM (Figure 9d), probably due to the poor blinking behaviour seen in the samples with the greater amount of dye, but further research is required.

Larger median values suggest a greater frequency of localisations per NP. The different formulations follow the same trend: more concentrated formulations show higher frequency for bigger amounts of dye (1 µM=66 loc/NP, 10 µM=231 loc/NP, 50 µM=1220 loc/NP and 100 µM=1512 loc/NP).
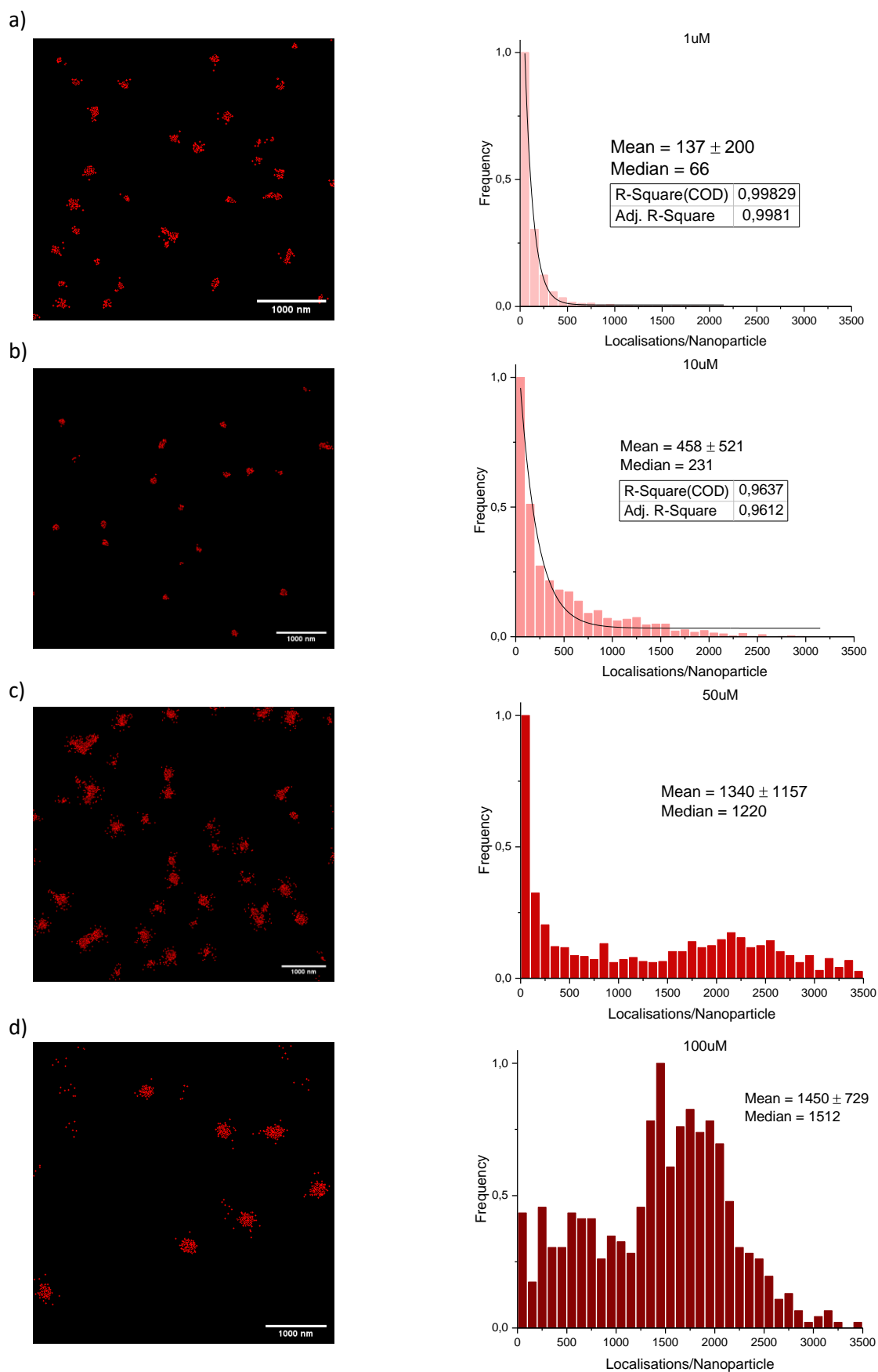
*Figure 9. Encapsulation efficiency for PLGA-PEG nanoparticles, with different concentrations of DiD (1 μM, 10 μM, 50 μM and 100 μM). Analysis performed in STORM files, using the Matlab clustering algorithm in section 10.2. **Left column:** STORM image, each red dot corresponds to a molecule of DiD. **Right column:** number of localisations per NP. The number of NPs analysed in each formulation was between 200-1000 NP. The results were then normalised.*

As observed in the size distribution analysis between the different formulations (manual nanoprecipitation and microfluidic device) in section 6.1, the process of formulating NPs is better controlled using the microfluidic device. For this reason, the encapsulation efficiency when formulating NPs manually may vary depending on other factors different than their size, such as the precision or the experience of the person formulating them. These limitations on manual formulations of NPs are also explored by Schubert et al., 2011.

On the other hand, despite the importance of determining the encapsulation efficiency, the current analytical methods are limited (Peng et al., 2016). For this reason, the use of STORM to quantify the encapsulation efficiency of the dye has been essential in order to obtain these results. Technically, this technique could also be applied to fluorescent therapeutics, such as doxorubicin (section 3.1. 1). It is important to note however, that not all fluorescent molecules have the same photo-switching properties, and moreover, the results depend on the Matlab parameters chosen by the user.
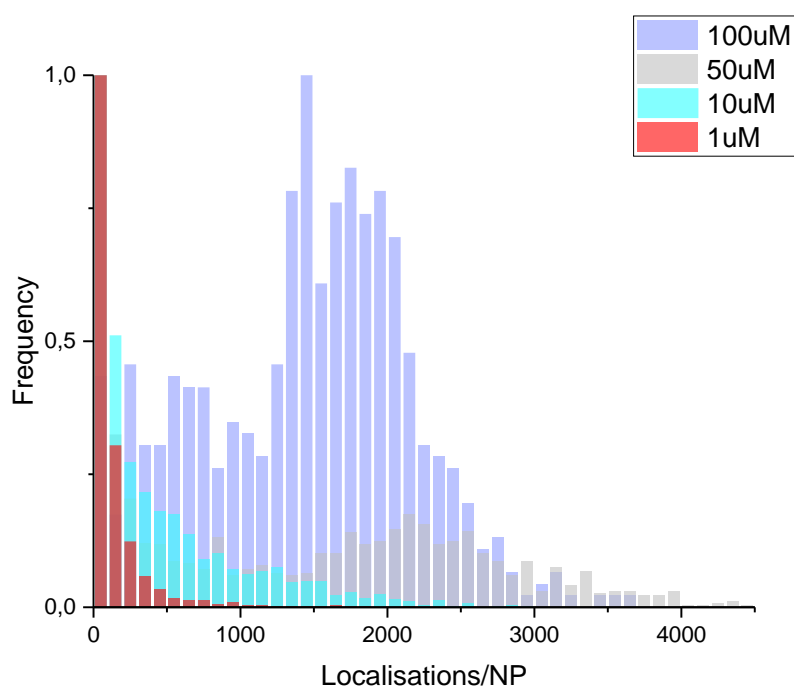
Figure 10. Encapsulation efficiency for PLGA-PEG nanoparticles, comparison of localisation per nanoparticle between different concentrations of DiD (1 μM, 10 μM, 50 μM and 100 μM). The number of NPs analysed in each formulation was between 200-1000 NP. The results were then normalised.

Figure 10 shows an overlay graph of the encapsulation efficiency results for the different concentrations of DiD. This way, the trend in the increasing concentration can be appreciated easily and it can be compared with another techniques used to measure the encapsulation efficiency, the ImageJ plugin ThunderSTORM. As seen in Figure 15 in the

annex, ThunderSTORM shows the same trend as Figure 9 but not as precisely since it is based in the intensity of the pixels rather than in the number of localisations. This ability to quantify the number of fluorophores at a single particle level of STORM gives it an advantage over the precision and quality of data compared to spectrophotometry measurements based on intensity.

Additionally, the encapsulation efficiency was also analysed between the two different formulation methods (manual nanoprecipitation and microfluidic device), which is shown in Figure 14 in the annex. The size distribution analysis of these formulations showed that formulating NPs using a microfluidic device provides more homogeneous and controlled samples, this trend is also seen with the encapsulation efficiency.

7. Conclusions

A robust characterisation of NPs is essential for a proper standardization and future approval for clinical applications. STORM has shown to be a more sensitive and precise characterisation method, both for quantifying the size and size distribution and the encapsulation efficiency of the NPs. In addition, comparing STORM to other analysis methods like DLS and TEM, has been useful to analyse NP both in dry state and in solution. Using these analysis techniques, it has been proved that the size, sample homogeneity and amount of dye encapsulated in the NPs is better controlled when they are formulated using a microfluidic device.

Furthermore, not only can STORM be used to measure NPs size, but it can also be used as a robust method to characterize dye encapsulation efficiency at single particle level. Working with different dye concentrations, it has been demonstrated that the encapsulation efficiency increases with the concentration of the dye encapsulated but at higher concentrations, inaccurate results are seen, probably due to the blinking behaviour of the dye at saturation point. However, a high concentration of dye is not normally used for NP encapsulation in literature.. Nonetheless, in comparison to more commonly used techniques, nanoparticle research using SRM methods such as STORM is still in its infancy and more research is required in this area for it to be implemented as a standard method.

In summary, STORM has proven to be a robust technique able to asses both size, size distribution and encapsulation efficiency of PLGA-PEG NP. Therefore, it shows great promise as an independent method, as well as in conjunction with standard techniques used for nanoparticle characterisation such as DLS and TEM, to gain unique insight and improve the overall understanding of physiochemical properties and cell-interaction behaviour of nanoparticles. For nanomedicine to reach its potential in the clinic, it is therefore necessary to encourage research in this field using more advanced techniques, such as super-resolution microscopy.

8. Future prospects

The advantages of using advanced microscopic techniques for nanomedicine research have been described in this paper. The next step in this research would therefore be to combine two of these powerful techniques to gain multi-dimensional information on the properties of NP, specifically through the use of CLEM, a state of the art technique able to combine the advantages of SRM and TEM. In this specific research, CLEM would allow a more in-depth study of the encapsulation efficiency analysis, comparing the size distribution of the NPs to the encapsulation efficiency directly. However, many challenges need to be overcome in order to achieve CLEM: STORM needs glass coverslips, which are impermeable to electrons, as well as an hydrated the samples while TEM uses dehydrated samples. Using TEM grids with the STORM buffer has been a challenge and still needs protocol optimization to handle the grids in different environments. Moreover, the sample preparation for TEM quenches fluorescence and causes background autofluorescence, for this reasons, the samples are first imaged in STORM. Finally, overlapping both TEM and STORM results is, as well, a laborious task since the same markers need to be in the picture and the sample is deformed during the processing steps.

Nevertheless, CLEM has been increasingly used in biological research and is showing great promise as an advanced technique in this area. But in spite of its success in biomedicine, CLEM has surprisingly not yet been applied to the study of nanomaterials. This opens a door of challenges and opportunities to effectively correlate different microscopic techniques to answer remaining questions in nanomedicine research.

9. References

Bhatia, Saurabh. 'Nanoparticles Types, Classification, Characterization, Fabrication Methods and Drug Delivery Applications'. *Natural Polymer Drug Delivery Systems*. Cham: Springer International Publishing, 2016: pp 33-93. https://doi.org/10.1007/978-3-319-41129-3.

Bretschneider, Stefan, Christian Eggeling, and Stefan W. Hell. 'Breaking the Diffraction Barrier in Fluorescence Microscopy by Optical Shelving'. *Physical Review Letters* 98, no. 21 (24 May 2007): 218103. https://doi.org/10.1103/PhysRevLett.98.218103.

Bulbake, Upendra, Sindhu Doppalapudi, Nagavendra Kommineni, and Wahid Khan. 'Liposomal Formulations in Clinical Use: An Updated Review'. *Pharmaceutics* 9, no. 4 (27 March 2017): 12. https://doi.org/10.3390/pharmaceutics9020012.

Clancy, B, and L.J Cauller. 'Reduction of Background Autofluorescence in Brain Sections Following Immersion in Sodium Borohydride'. *Journal of Neuroscience Methods* 83, no. 2 (September 1998): 97–102. https://doi.org/10.1016/S0165-0270(98)00066-1.

Cohen, Arthur L. 'Electron Microscopy: Principles and Techniques for Biologists. John J. Bozzola , Lonnie D. Russell'. *The Quarterly Review of Biology* 67, no. 2 (1 June 1992): 247–49. https://doi.org/10.1086/417649.

Danhier, Fabienne, Eduardo Ansorena, Joana M. Silva, Régis Coco, Aude Le Breton, and Véronique Préat. 'PLGA-Based Nanoparticles: An Overview of Biomedical Applications'. *Journal of Controlled Release* 161, no. 2 (July 2012): 505–22. https://doi.org/10.1016/j.jconrel.2012.01.043.

Date, Abhijit A., and M.S. Nagarsenker. 'Parenteral Microemulsions: An Overview'. *International Journal of Pharmaceutics* 355, no. 1–2 (1 May 2008): 19–30. https://doi.org/10.1016/j.ijpharm.2008.01.004.

Dempsey, Graham T., Joshua C. Vaughan, Kok Hao Chen, Mark Bates, and Xiaowei Zhuang. 'Evaluation of Fluorophores for Optimal Performance in Localization-Based Super-Resolution Imaging'. *Nature Methods* 8, no. 12 (6 November 2011): 1027–36. https://doi.org/10.1038/nmeth.1768.

Ehmann, Nadine, Sebastian van de Linde, Amit Alon, Dmitrij Ljaschenko, Xi Zhen Keung, Thorge Holm, Annika Rings, et al. 'Quantitative Super-Resolution Imaging of Bruchpilot Distinguishes Active Zone States'. *Nature Communications* 5, no. 1 (December 2014): 4650. https://doi.org/10.1038/ncomms5650.

Faria, Matthew, Mattias Björnmalm, Kristofer J. Thurecht, Stephen J. Kent, Robert G. Parton, Maria Kavallaris, Angus P. R. Johnston, et al. 'Minimum Information Reporting in Bio–Nano Experimental Literature'. *Nature Nanotechnology* 13, no. 9 (September 2018): 777–85. https://doi.org/10.1038/s41565-018-0246-4.

Golombek, Susanne K., Jan-Niklas May, Benjamin Theek, Lia Appold, Natascha Drude, Fabian Kiessling, and Twan Lammers. 'Tumor Targeting via EPR: Strategies to Enhance Patient Responses'. *Advanced Drug Delivery Reviews* 130 (May 2018): 17–38. https://doi.org/10.1016/j.addr.2018.07.007.

Grogger, Werner, Bernhard Schaffer, Kannan M. Krishnan, and Ferdinand Hofer. 'Energy-Filtering TEM at High Magnification: Spatial Resolution and Detection Limits'. *Ultramicroscopy* 96, no. 3–4 (September 2003): 481–89. https://doi.org/10.1016/S0304-3991(03)00110-4.

Gulati, Neha and Himanshu Gupta. 'Parenteral drug delivery: a review'. *Recent Patents on Drug Delivery & Formulation* , vol. 5, no. 2 (May 2011): 133–145. https://doi.org/10.2174/187221111795471391.

Ha, Taekjip, and Philip Tinnefeld. 'Photophysics of Fluorescent Probes for Single-Molecule Biophysics and Super-Resolution Imaging'. *Annual Review of Physical Chemistry* 63, no. 1 (5 May 2012): 595–617. https://doi.org/10.1146/annurev-physchem-032210-103340.

Hauser, Meghan, Michal Wojcik, Doory Kim, Morteza Mahmoudi, Wan Li, and Ke Xu. 'Correlative Super-Resolution Microscopy: New Dimensions and New Opportunities'. *Chemical Reviews* 117, no. 11 (14 June 2017): 7428–56. https://doi.org/10.1021/acs.chemrev.6b00604.

Hell, S.W. & Kroug, M. Appl. Phys. B Ground-state-depletion fluorscence microscopy: A concept for breaking the diffraction resolution limit. *Applied Physics B* 60, no 5 (10 March 1995) 495-497. https://doi.org/10.1007/BF01081333.

Hell, Stefan W., and Jonathan Wichmann. 'Breaking the Diffraction Resolution Limit by Stimulated Emission: Stimulated-Emission-Depletion Fluorescence Microscopy.' *Optics Letters* 19, no. 11 (1994): 780–82. https://doi.org/10.1364/ol.19.000780.

Introduction to Fluorescence Microscopy | MicroscopyU. Accessed September 2019. https://www.microscopyu.com/techniques/fluorescence/introduction-to-fluorescence-microscopy.

John R. Allen, Joel S. Silfies and Stanley A. Schwartz and Michael W. Davidson 'Single-Molecule Super-Resolution Imaging'. *Nikon Instruments, Inc.* (2019): Microscopy U.

Jungmann, Ralf, Maier S Avendaño, Mingjie Dai, Johannes B Woehrstein, Sarit S Agasti, Zachary Feiger, Avital Rodal, and Peng Yin. 'Quantitative Super-Resolution Imaging with QPAINT Using Transient Binding Analysis'. *Nature Methods* 13, no. 5 (May 2016): 439–42. https://doi.org/10.1038/nmeth.3804.

Khan, Ibrahim, Khalid Saeed, and Idrees Khan. 'Nanoparticles: Properties, Applications and Toxicities'. *Arabian Journal of Chemistry* 12, no. 7 (November 2019): 908–31. https://doi.org/10.1016/j.arabjc.2017.05.011.

Kim, Doory, Thomas J. Deerinck, Yaron M. Sigal, Hazen P. Babcock, Mark H. Ellisman, and Xiaowei Zhuang. 'Correlative Stochastic Optical Reconstruction Microscopy and Electron Microscopy'. Edited by Michael A. Fox. *PLOS ONE* 10, no. 4 (15 April 2015): e0124581. https://doi.org/10.1371/journal.pone.0124581.

Klar, T. A., S. Jakobs, M. Dyba, A. Egner, and S. W. Hell. 'Fluorescence Microscopy with Diffraction Resolution Barrier Broken by Stimulated Emission'. *Proceedings of the National Academy of Sciences* 97, no. 15 (18 July 2000): 8206–10. https://doi.org/10.1073/pnas.97.15.8206.

Kumari, Avnesh, Sudesh Kumar Yadav, and Subhash C. Yadav. 'Biodegradable Polymeric Nanoparticles Based Drug Delivery Systems'. *Colloids and Surfaces B: Biointerfaces* 75, no. 1 (January 2010): 1–18. https://doi.org/10.1016/j.colsurfb.2009.09.001.

Maguire, Ciarán Manus, Matthias Rösslein, Peter Wick, and Adriele Prina-Mello. 'Characterisation of Particles in Solution – a Perspective on Light Scattering and Comparative Technologies'. *Science and Technology of Advanced Materials* 19, no. 1 (31 December 2018): 732–45. https://doi.org/10.1080/14686996.2018.1517587.

Malvern Instruments 'Dynamic Light Scattering: An Introduction in 30 Minutes'. *Technical note* (Accessed August 2019). https://www.malvernpanalytical.com/en/learn/knowledge-center/technical-notes/TN101104DynamicLightScatteringIntroduction.

Monticone, D. Gatto, K. Katamadze, P. Traina, E. Moreva, J. Forneris, I. Ruo Berchera, P. Olivero, I. P. Degiovanni, G. Brida, and M. Genovese. 'Beating Abbe Diffraction Limit in Confocal Microscopy via Non-Classical Photon Statistics'. *Physical Review Letters* 113, no. 14 (30 September 2014): 143602. https://doi.org/10.1103/PhysRevLett.113.143602.

Owens III, D, and N Peppas. 'Opsonization, Biodistribution, and Pharmacokinetics of Polymeric Nanoparticles'. *International Journal of Pharmaceutics* 307, no. 1 (3 January 2006): 93–102. https://doi.org/10.1016/j.ijpharm.2005.10.010.

Peng, Zhili, Shanghao Li, Xu Han, Abdulrahman O. Al-Youbi, Abdulaziz S. Bashammakh, Mohammad S. El-Shahawi, and Roger M. Leblanc. 'Determination of the Composition, Encapsulation Efficiency and Loading Capacity in Protein Drug Delivery Systems Using Circular Dichroism Spectroscopy'. *Analytica Chimica Acta* 937 (September 2016): 113–18. https://doi.org/10.1016/j.aca.2016.08.014.

Pujals, Silvia, Natalia Feiner-Gracia, Pietro Delcanale, Ilja Voets, and Lorenzo Albertazzi. 'Super-Resolution Microscopy as a Powerful Tool to Study Complex Synthetic Materials'. Nature Reviews Chemistry 3, no. 2 (February 2019): 68–84. https://doi.org/10.1038/s41570-018-0070-2.

Sanderson, M. J., I. Smith, I. Parker, and M. D. Bootman. 'Fluorescence Microscopy'. *Cold Spring Harbor Protocols* 2014, no. 10 (1 October 2014): pdb.top071795-pdb.top071795. https://doi.org/10.1101/pdb.top071795.

Sauer, Markus, and Mike Heilemann. 'Single-Molecule Localization Microscopy in Eukaryotes'. *Chemical Reviews* 117, no. 11 (14 June 2017): 7478–7509. https://doi.org/10.1021/acs.chemrev.6b00667.

Schermelleh, Lothar, Alexia Ferrand, Thomas Huser, Christian Eggeling, Markus Sauer, Oliver Biehlmaier, and Gregor P. C. Drummen. 'Super-Resolution Microscopy Demystified'. *Nature Cell Biology* 21, no. 1 (January 2019): 72–84. https://doi.org/10.1038/s41556-018-0251-8.

Schoen, Ingmar, Jonas Ries, Enrico Klotzsch, Helge Ewers, and Viola Vogel. 'Binding-Activated Localization Microscopy of DNA Structures'. *Nano Letters* 11, no. 9 (14 September 2011): 4008–11. https://doi.org/10.1021/nl2025954.

Schubert, Stephanie, Joseph T. Delaney Jr and Ulrich S. Schubert. 'Nanoprecipitation and nanoformulation of polymers: from history to powerful possibilities beyond poly(lactic acid)'. *Soft Matter 7* (2011): 1581-1588. https://doi.org/10.1039/C0SM00862A.

Sheela A. Abraham, Dawn N. Waterhouse, Lawrence D. Mayer, Pieter R. Cullis, Thomas D. Madden, Marcel B. Bally. 'The Liposomal Formulation of Doxorubicin'. *Methods in Enzymology*, Volume 391 (2005): 71-97. https://doi.org/10.1016/S0076-6879(05)91004-5 .

Singh, Rajesh, and James W. Lillard. 'Nanoparticle-Based Targeted Drug Delivery'. *Experimental and Molecular Pathology* 86, no. 3 (June 2009): 215–23. https://doi.org/10.1016/j.yexmp.2008.12.004.

Stetefeld, Jörg, Sean A. McKenna, and Trushar R. Patel. 'Dynamic Light Scattering: A Practical Guide and Applications in Biomedical Sciences'. *Biophysical Reviews* 8, no. 4 (December 2016): 409–27. https://doi.org/10.1007/s12551-016-0218-6.

Szczurek, Aleksander, Ludger Klewes, Jun Xing, Amine Gourram, Udo Birk, Hans Knecht, Jurek W. Dobrucki, Sabine Mai, and Christoph Cremer. 'Imaging Chromatin Nanostructure with Binding-Activated Localization Microscopy Based on DNA Structure Fluctuations'. *Nucleic Acids Research* 45, no. 8 (5 May 2017): e56. https://doi.org/10.1093/nar/gkw1301.

Tang, C.Y., and Z. Yang. 'Transmission Electron Microscopy (TEM)'. In *Membrane Characterization*, 145–59. Elsevier, 2017. https://doi.org/10.1016/B978-0-444-63776-5.00008-5.

Ventola, C Lee. 'Progress in Nanomedicine: Approved and Investigational Nanodrugs', n.d., 14 (December 2017).

W. Davidson, Michael and Mortimer Abramowitz 'Optical Microscopy'. *In Encyclopedia of Imaging Science and Technology, J.P. Hornak (Ed.)* (2002): 777–85. https://doi.org/10.1002/0471443395.img074.

Watanabe, Shigeki, Annedore Punge, Gunther Hollopeter, Katrin I Willig, Robert John Hobson, M Wayne Davis, Stefan W Hell, and Erik M Jorgensen. 'Protein Localization in Electron Micrographs Using Fluorescence Nanoscopy'. *Nature Methods* 8, no. 1 (January 2011): 80–84. https://doi.org/10.1038/nmeth.1537.

Wilhelm, Stefan, Anthony J. Tavares, Qin Dai, Seiichi Ohta, Julie Audet, Harold F. Dvorak, and Warren C. W. Chan. 'Analysis of Nanoparticle Delivery to Tumours'. *Nature Reviews Materials* 1, no. 5 (May 2016): 16014. https://doi.org/10.1038/natrevmats.2016.14.

Winey, Mark, Janet B. Meehl, Eileen T. O'Toole, and Thomas H. Giddings. 'Conventional Transmission Electron Microscopy'. *Molecular Biology of the Cell* 25, no. 3 (1 February 2014): 319–23. https://doi.org/10.1091/mbc.E12-12-0863.

Yadav, Khushwant S., and Krutika K. Sawant. 'Modified Nanoprecipitation Method for Preparation of Cytarabine-Loaded PLGA Nanoparticles'. *AAPS PharmSciTech* 11, no. 3 (September 2010): 1456–65. https://doi.org/10.1208/s12249-010-9519-4.

Zwaag, Daan van der, Nane Vanparijs, Sjors Wijnands, Riet De Rycke, Bruno G. De Geest, and Lorenzo Albertazzi. 'Super Resolution Imaging of Nanoparticles Cellular Uptake and Trafficking'. *ACS Applied Materials & Interfaces* 8, no. 10 (16 March 2016): 6391–99. https://doi.org/10.1021/acsami.6b00811.

10.   Self-evaluation

Personally, I consider this research project as the best experience I have had during my degree. Thanks to this experience, I was able to completely integrate in a research group with such an important focus of research, which is nanomedicine. I believe the project I have been working on to characterise NP is quite relevant to the standardization and future clinical application of nanoparticles in the medical field. Moreover, working with such complex and advanced techniques such as SRM and TEM has been a unique and enriching experience. Furthermore, having trained with these techniques at such an early stage of my research experience will benefit my career progression in the long-term.

Additionally, in the months I stayed in the laboratory I have had the opportunity to learn the protocols to the point of becoming completely independent in the experimental work and design of a protocol, as well as improving my time management and writing skills. Moreover, I have been able of unifying both degrees I completed in the last five years, Computer Engineering and Biotechnology. I have seen the importance of the Computer Engineering in Bioinformatics in order to analyse big amounts of data, for example, when using Matlab for STORM analysis. In addition, I feel extremely lucky for being able to develop a new Matlab application for the correlation of TEM and STORM images, as part of the Computer Engineering research project, in the same research group. This way, I can really merge the knowledge I gained in my degrees with the same research purpose.

Finally, I am thankful for my workmates, who have helped me during my learning process and who have answered any doubts I had. Especially to my supervisors, Teodora Andrian and Dr. Silvia Pujals, and to the group leader Dr. Lorenzo Albertazzi.

11.   Annexes

11.1.   Protocols

[CLEM] STORM on PLGA-PEG and Gold (Au) NPs preparation protocol:

1. Prepare dilutions of PLGA-PEG NPs in filtered PBS and gold NPs as fiducial markers
   a. Vortex everything before use; PLGA-PEG NP and AuNP for 1min.
   b. Up&down pipetting.
2. Prepare Glox Buffer.
3. UV Glow Discharge 30s on TEM grids
4. Attach diluted NPs to grids (~40µl drop) during 10min (drop on bottom) and gold NPs with the drop on top. Wash for 1min with a drop of filtered PBS.
5. Wash microscope slide with ethanol, place grid and a drop of 40µl of glox buffer. Place coverslip and take out excess with a tissue.
6. STORM image acquisition.
7. Take grid out of the microscope slide placing it in PBS and keep it for TEM image acquisition.

[CLEM] TEM on PLGA-PEG and gold (Au) NPs preparation protocol:

1. Get grids prepared following the [CLEM] STORM protocol.
2. Negative staining:
   a. Wash 3 times with filtered MilliQ water (1min, 30s and 30s), placing the grid on a ~40µl drop.
   b. Place grid on a Uranyl Acetate 2% drop during 1min.

11.2. Matlab analysis for STORM text files scripts

ReadCoords.m

```matlab
function
[XCoords647,YCoords647,TCoords647,XCoordsFid,YCoordsFid,TCoordsFid]=
ReadCoords(FileName, InputType, varargin)

        %%%%%   READCOORDS: reads raw data from SMLM
        %%%%% read raw data (TXT or CSV) obtained from SMLM analysis
in
        %%%%% NIS-elements (Nikon N-STORM) or ONI software, and
extract
        %%%%% coordinates of interest for further processing. It
generates
        %%%%% txt file of 3 columns containing XYT coordinates to
use for
        %%%%% further processing.
        %%%%%

        %-----------------------------------------------------------
-------
        % INPUTS:
        % FileName: name of the file with extension, e.g.
'MyFile.txt'
        % InputType: denotes the type of file, type 'N-STORM' for
Nikon
        % software TXT files, or 'ONI' for CSV files from ONI.
        %
        % N.B.
        % only two-channels can be read: main-channel (named 647)
and
        % fiducial markers channels (named Fid). File format should
be
        % checked.
        %
        % N-STORM files (TXT) are supposed to be 26-column, columns
4-5-13 are
        % read as X-Y-T. !! by default the function splits channels
        % according to first column (named Channel Name) in TXT
file,
        % fiducial coords are for 'Bead Drift Correction', 647-
coords for
        % '647'!!
        %
        % ONI files (CSV) are supposed to be 11-columns, columns 3-
4-2 are
        % read as X-Y-T. !! by default the function splits channels
        % according to first column (named Channel) in CSV file,
fiducial
        % coords are for Channel=0, 647-coords for Channel=1.!!
        %
        %-----------------------------------------------------------
-------
        % OUTPUTS:
        % X-Y-TCoords647: X, Y, T(frames) coordinates of
localization in
        % the main channel, named 647, expressed in nanometers
        %
        % X-Y-TCoordsFid: X, Y, T(frames) coordinates of
localization in
```

```matlab
        % the second channel (typically fiducial markers), expressed
in
        % nanometers
        %
        %
        %------------------------------------------------------------
-------

p = inputParser; %init parser object
validChar = @(x) ischar(x);
validNum = @(x) isinteger(x);
% here add something to read the coordinate file


%define defaults values for optional param:
defaultSTORMname = '647'; % main channel name for N-STORM data
defaultONIname = 1 ; % main channel name for ONI
defaultSTORMref = 'Bead Drift Correction'; % ref channel name for N-
STORM data
defaultONIref = 0 ; % ref channel name for ONI

%define required and optional input parameters:
addRequired(p,'FileName',validChar);
addRequired(p,'InputType',validChar);

addParameter(p,'STORMname', defaultSTORMname, validChar);
addParameter(p,'STORMref', defaultSTORMref, validChar);
addParameter(p,'ONIname', defaultONIname, validNum);
addParameter(p,'ONIref', defaultONIref, validNum);

%read input values:
parse(p, FileName, InputType, varargin{:});

%assign the parsed values:
FileName = p.Results.FileName; % file name
InputType = p.Results.InputType;
STORMname = p.Results.STORMname;
STORMref = p.Results.STORMref;
ONIname = p.Results.ONIname;
ONIref = p.Results.ONIref;

switch InputType


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%
    %%%%------ TXT file from N-STORM -------------------------------
-------

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%

    case 'N-STORM'

        %%%----- Read text file (check format)---
        disp('Importing N-STORM data...');
        fileID = fopen(FileName,'r');
        DataIn = textscan(fileID,'%s %f %f %f %f %f %f %f %f %f %f
%f %f %f %f %f %f %f %f %f %f %f %f
%f','Delimiter','\t','HeaderLines',1); % Saving data as a cell,
check the numb of columns
```

```matlab
        fclose(fileID);

        %%%--- Retrieve useful information (channel, corrected XY
location and frame number or the localization)
        XCoords = cell2mat(DataIn(4)); %% colum 2 X non drift
corrected, 4 X drift corrected
        YCoords = cell2mat(DataIn(5)); %% colum 3 Y non drift
corrected, 5 Y drift corrected
        TCoords = cell2mat(DataIn(13)); %% column 13 Frame
        Channel = DataIn(1);
        clear DataIn;

        %%%--- Split channel based on first column and generate
output
        %%%--- main channel is 647 (first column of TXT file)
        Channel647=strcmp(Channel{1},STORMname);%Check in the .txt
if 647 or 405/647
        XCoords647=XCoords(Channel647);
        YCoords647=YCoords(Channel647);
        TCoords647=TCoords(Channel647);
        ChannelFid=strcmp(Channel{1},STORMref);%Check in the .txt if
647 or other name
        XCoordsFid=XCoords(ChannelFid);
        YCoordsFid=YCoords(ChannelFid);
        TCoordsFid=TCoords(ChannelFid);

        %%%--- Export a txt file with XYT coords
        n=length(XCoords647);
        nref=length(XCoordsFid);
        A=zeros(n,3);
        Ref=zeros(nref,3);
        A(:,1)=XCoords647;
        A(:,2)=YCoords647;
        A(:,3)=TCoords647;
        Ref(:,1)=XCoordsFid;
        Ref(:,2)=YCoordsFid;
        Ref(:,3)=TCoordsFid;
        save XYTcoordinates.txt A -ascii
        save XYTref.txt Ref -ascii


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%
    %%%%------ CSV file from ONI ----------------------------------
-------

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%

    case 'ONI'

        %%%----- Read text file (check format)---
        disp('Importing ONI data...');
        fileID = fopen(FileName,'r');
        DataIn = textscan(fileID,'%f %f %f %f %f %f %f %f %f %f
%f','Delimiter',',','HeaderLines',1); % Saving data as a cell, check
the numb of columns
        fclose(fileID);

        %%%--- Retrieve useful information (channel, corrected XY
location and frame number or the localization)
```

```matlab
        XCoords = cell2mat(DataIn(3)); %% colum 3 X coords
        YCoords = cell2mat(DataIn(4)); %% colum 4 Y coords
        TCoords = cell2mat(DataIn(2)); %% column 2 Frame coords
        Channel = DataIn(1);
        clear DataIn;


        %%%--- Split channel based on first column and generate
output
        Channel647= (Channel{1}==ONIname); % 647-channel  with 1
        XCoords647=XCoords(Channel647);
        YCoords647=YCoords(Channel647);
        TCoords647=TCoords(Channel647);
        ChannelFid= (Channel{1}==ONIref);%C Fiducial-channels
denoted with 0
        XCoordsFid=XCoords(ChannelFid);
        YCoordsFid=YCoords(ChannelFid);
        TCoordsFid=TCoords(ChannelFid);

        %%%--- Export a txt file with XYT coords
        n=length(XCoords647);
        nref=length(XCoordsFid);
        A=zeros(n,3);
        Ref=zeros(nref,3);
        A(:,1)=XCoords647;
        A(:,2)=YCoords647;
        A(:,3)=TCoords647;
        Ref(:,1)=XCoordsFid;
        Ref(:,2)=YCoordsFid;
        Ref(:,3)=TCoordsFid;
        save XYTcoordinates.txt A -ascii
        save XYTref.txt Ref -ascii


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%
    %%%%------ CSV file from Thunderstorm ------------------------
-------

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%

%     case 'Thunder'
%
%         %%%----- Read text file (check format)---
%         disp('Importing Thunder data...');
%         fileID = fopen(FileName,'r');
%         DataIn = textscan(fileID,'%f %f %f %f %f %f
%f','Delimiter',',','HeaderLines',1); % Saving data as a cell, check
the numb of columns
%         fclose(fileID);
%
%         %%%--- Retrieve useful information (channel, corrected XY
location and frame number or the localization)
%         XCoords647 = cell2mat(DataIn(3));
%         YCoords647 = cell2mat(DataIn(4));
%         TCoords647 = cell2mat(DataIn(2));
%         XCoordsFid = 0;
%         YCoordsFid = 0;
%         TCoordsFid = 0;
%         %Channel = DataIn(1);
```

```matlab
%          clear DataIn;
%
%
%          %%%--- Split channel based on first column and generate
output
%          Channel647= (Channel{1}==1); % 647-channel denoted with 1
%          XCoords647=XCoords(Channel647);
%          YCoords647=YCoords(Channel647);
%          TCoords647=TCoords(Channel647);
%          ChannelFid= (Channel{1}==0);%C Fiducial-channels denoted
with 0
%          XCoordsFid=XCoords(ChannelFid);
%          YCoordsFid=YCoords(ChannelFid);
%          TCoordsFid=TCoords(ChannelFid);
%
    otherwise
        disp('invalid InputType!');
end
end
```

Cluster.m

```matlab
function [ClustSize, diam, Loc2particle]=Cluster(FileName,
Bandwidth, MinPts, MaxDiam, varargin)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%
%%%%% CLUSTER: perform clustering of (X,Y,T) coordinates with Mean-
Shift
%%%%% algorithm using a SINGLE channel to identify clusters having
%%%%% particle-like shape with user defined selection parameters.
%%%%% REQUIRED INPUTS:
%%%%% file name: XYT coords, in nm and frame number
%%%%% bandwidth: parameter for clustering (in nm)
%%%%% MinPts: minimum number of localiz in a cluster
%%%%% MaxDiam: maximum diameter (longest axis) in nm
%%%%% OUTPUTS:
%%%%% Loc2particle: cell array with all info for each selected
cluster
%%%%% ClusSize: number of localiz in each selected cluster
%%%%% diam: diameter of each selected cluster (in nm)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%

p = inputParser; %init parser object
validNum = @(x) isnumeric(x) && (x > 0); %define valid inputs:
positive num
validChar = @(x) ischar(x);
% here add something to read the coordinate file


%define defaults values for optional param:
defaultElong = 2.0; %elongation in ellipse fit
defaultScaleFactor= 1.0; %scale factor in ellipse fit
defaultMinClustDist = 300; % min distance between clusters
defaultAggrDist = defaultMinClustDist; % min distance between
clusters to be non-aggregate, by default like MinClustDist

%define required and optional input parameters:
```

```matlab
addRequired(p,'FileName',validChar);
addRequired(p,'Bandwidth',validNum);
addRequired(p,'MinPts',validNum);
addRequired(p,'MaxDiam', validNum);

addParameter(p,'Elong', defaultElong, validNum);
addParameter(p,'ScaleFactor', defaultScaleFactor, validNum);
addParameter(p,'MinClustDist', defaultMinClustDist, validNum);
addParameter(p,'AggrDist', defaultAggrDist, validNum);

%read input values:
parse(p,FileName, Bandwidth, MinPts, MaxDiam,varargin{:});

%assign the parsed values:
Last = Inf; % Limit the number of points processed (set to Inf to
process all)
MaxParticleElongation = p.Results.Elong; % max elongation allowed in
ellipse fit
EllipseFitScaleFactor = p.Results.ScaleFactor; % scale factor in
ellipse fit
MinClustDst = p.Results.MinClustDist; % min distance of closest
cluster to be considered isolated
distanceAggregates = p.Results.AggrDist; % min distance between
clusters to be non-aggregate

% % if not introduced, by default distance aggregate is set as
MinClustDst:
% if (MinClustDst ~= defaultMinClustDist) && (distanceAggregates ==
defaultAggrDist)
%     distanceAggregates = defaultMinClustDist;
%

% read XYT coords in txt file:
Coordinates=importdata(FileName);
% %%read txt with more complex file structure:
% % delimiterIn = ' ';
% % headerlinesIn = 1;
% % Coordinates = importdata(FileName,delimiterIn,headerlinesIn);

% assign X-Y-T coordinates:
XCoords647 = Coordinates(:,1); %X coords in first column
YCoords647 = Coordinates(:,2); %Y coords in second column
TCoords647 = Coordinates(:,3); %T coords in third column

% Plot raw coordinates data:
XCoords647 = XCoords647(1:min(Last,numel(XCoords647))); %numel
Number of elements in an array or subscripted array expression
YCoords647 = YCoords647(1:min(Last,numel(YCoords647)));
plot(XCoords647,YCoords647,'.r'); axis equal; hold on;

% Clustering using mean-shift:
disp('Clustering...');
Pts647 = [XCoords647 YCoords647];
[clustCent647,~,clustMembsCell647] =
MeanShiftCluster(Pts647.',Bandwidth); % .' transpose of the matrix
ClustSize647 = cellfun(@numel, clustMembsCell647);
NClust647 = numel(clustMembsCell647);
disp(strcat(['Found ' num2str(NClust647) ' clusters']));

% Flag isolated Nanoparticles
```

```matlab
[D, ~] =
pdist2(clustCent647',clustCent647','euclidean','Smallest',2);
isolated647 = (D(2,:) >= MinClustDst);


% Loop: Fit ellipse, then elongation filter + size filter  --> out:
 %this removes clusters with few localiz, and too elongated shape

IndexRightClust647=false(NClust647,1); % Pietro: init logical array
to flag valid clusters
RightClust647=1:1:NClust647; % Pietro: intialization cluster indexes
massCenter=clustCent647; %Pietro: initialization centers coords of
clusters
r=0;
for i = 1 : NClust647
    A = [XCoords647(cell2mat(clustMembsCell647(i)))';
YCoords647(cell2mat(clustMembsCell647(i)))' ];
    if(size(A,2) >= MinPts)
        % Filter particle based on ellipse elongation & major axis
length
        % Plot ellipse as reference
        ellipse_t =
fit_ellipse(A(1,:),A(2,:),gcf,EllipseFitScaleFactor,MaxParticleElong
ation,MaxDiam, isolated647(i));
        valid = ellipse_t.valid;
        if valid == 1
            IndexRightClust647(i)=true; %Pietro: flag as true the
valid cluster (numb points and elongation)
        end
    else
      disp(strcat( ['#' num2str(i) ' cluster has few localizations
and has been excluded']));
      r=r+1; %count discarded clusters
    end
end

RightClust647=RightClust647(IndexRightClust647); % Pietro: indexes
of valid clusters (numb points and elongation)
massCenter=massCenter(:,IndexRightClust647); % Pietro: centers of
valid clusters (numb points and elongation)
%ClustSize647=ClustSize647(IndexRightClust647); %Pietro: size of
valid clusters

disp(strcat( [num2str(NClust647-length(RightClust647)-r) ' clusters
have been excluded due to unrealistic size or elongation']));
hold on
plot(massCenter(1,:), massCenter(2,:), 'xk', 'LineWidth',3,
'MarkerSize',10);
%Pietro: this is to visualize the label of the identified clusters
for i = 1 : length(massCenter)
txt1 = ['\leftarrow ' num2str(i)];
text(massCenter(1,i),massCenter(2,i),txt1)
end

%% Filter particle too close /aggregates
distanceCenters=pdist2(massCenter',massCenter');
[rows,~]=find((distanceCenters~=0) &
(distanceCenters<distanceAggregates));
AggregateMember= rows';
Aggregates=RightClust647(AggregateMember);
```

```matlab
AggrMembr=clustMembsCell647(Aggregates); %for every aggregate which
points are in it
RightClust647(AggregateMember)=[];
NPMembs=clustMembsCell647(RightClust647); %for every nanoparticle
which points are in it
NPSize647 = cellfun(@numel, NPMembs);  % added by Pietro: for each
selected NP, counts the number of points in it

disp(strcat( [num2str(length(AggregateMember)) ' clusters have been
excluded because were forming aggregates']));
disp(strcat(['Identified ' num2str(length(RightClust647)) '
nanoparticles candidates from ' num2str(NClust647) ' candidate
clusters']));

figure(2)
hold on
for i=1: length(Aggregates)
    AggregateData=[XCoords647(cell2mat(AggrMembr(i))),
YCoords647(cell2mat(AggrMembr(i)))];
    plot(AggregateData(:,1), AggregateData(:,2), 'xy')
end


% Size Check
%this performs a circle-fitting of the clusters to calculate the
radius
%comprising a fraction of the localization (! remember to adjust the
values
%Fracthreshold and Rcheck high and low!), finally NPs with
unrealistic size
%are discarded


DataType='SolidSphere';
nclusters=length(RightClust647);
%SizeCheck=zeros(nclusters,1);
C=zeros(nclusters,2);
R=zeros(nclusters,1);
Rcheck=zeros(nclusters,1);
RCheckLow=5;
RCheckHigh=400;
discard=0;
IndexToRemove=true(nclusters,1); % Pietro: logical array to remove
index of cluster discarded
for i=1:nclusters
    switch DataType
        %case 'HollowSphere'
            %
[C(i,1),C(i,2),R(i)]=circfit(StoreClusterCoords{i,1}(:,1),StoreClust
erCoords{i,1}(:,2));
        case 'SolidSphere'
            ClusterData=[XCoords647(cell2mat(NPMembs(i))),
YCoords647(cell2mat(NPMembs(i)))];
            %plot(ClusterData(:,1), ClusterData(:,2), 'xg')
            Cinitial=[mean(ClusterData(:,1))
mean(ClusterData(:,2))];
            FracThreshold=0.90;
            % As an initial guess for the radius, take the radius
that
            % includes 90% of the datapoints. This is about the
distance of
```

```matlab
            % 1.5x the standard deviation, which is crudely
approximated
            % here by the average of the sxx, syy and szz. This
assumes a
            % spherical point distribution. A more accurate
            % description can be gained by using the full covariance
matrix
            % and computing the Mahalanobis distance.
            CovMat=cov(ClusterData);
            Rinitial=1.5*mean([sqrt(CovMat(1,1))
sqrt(CovMat(2,2))]);

            % Now the location of the center of the smallest sphere
            % encompassing 90% of the datapoints is determined.
            CenterSampleSizeAz=10; CenterSampleSizeRad=4;

CenterLocation=zeros(CenterSampleSizeAz+1,2,CenterSampleSizeRad);

TrackLocNumber=zeros(CenterSampleSizeAz+1,CenterSampleSizeRad);

            for n=1:CenterSampleSizeRad
                t2 = linspace(0,2*pi,CenterSampleSizeAz);
                XCurrentCircle=0.05*n*Rinitial*cos(t2)+Cinitial(1);
                YCurrentCircle=0.05*n*Rinitial*sin(t2)+Cinitial(2);
                for o1=1:CenterSampleSizeAz
                        CenterLocation(o1,:,n)=[XCurrentCircle(o1)
YCurrentCircle(o1)];
                        LocDistCheck=find(((ClusterData(:,1)-
XCurrentCircle(o1)).^2+(ClusterData(:,2)-YCurrentCircle(o1)).^2) <
Rinitial^2);
                        TrackLocNumber(o1,n)=length(LocDistCheck);
                end
            end

            RefLocCheck=find(((ClusterData(:,1)-
Cinitial(1)).^2+(ClusterData(:,2)-Cinitial(2)).^2) < Rinitial^2);
            RefLocNumber=length(RefLocCheck);
            MaxLoc=max(max(TrackLocNumber));
            if MaxLoc > RefLocNumber
                [I,J]=find(TrackLocNumber==MaxLoc);
                MaxLocMinRadTemp=[I J];
                MaxLocMinRadInd=find(J==min(J));
                MaxLocMinRad=MaxLocMinRadTemp(MaxLocMinRadInd,:);

Cfinal=mean(CenterLocation(MaxLocMinRad(:,1),:,min(J)),1);
            else
                Cfinal=Cinitial;
            end

            RadiusVec=linspace(0,3*Rinitial,1500);
            RadiusNumLoc=zeros(length(RadiusVec),1);
            for p=1:length(RadiusVec)
                LocDistCheckRad=find(((ClusterData(:,1)-
Cfinal(1)).^2+(ClusterData(:,2)-Cfinal(2)).^2) < RadiusVec(p)^2);
                RadiusNumLoc(p)=length(LocDistCheckRad);
            end

            RadTrack=1;
            while RadiusNumLoc(RadTrack) < FracThreshold *
size(ClusterData,1)
                RadTrack=RadTrack+1;
```

```matlab
            end
            Rfinal=RadiusVec(RadTrack);

            C(i,:)=Cfinal';
            Rcheck(i)=Rfinal;
    end
    if Rcheck(i) >= RCheckLow && Rcheck(i) <= RCheckHigh
        R(i)=Rfinal;
    else
        disp(['Cluster #' num2str(i) ' has been excluded from the
analysis due to an unrealistic size:' num2str(round(Rcheck(i))*2)])
        IndexToRemove(i) = false; % Pietro: collect the indexes of
the clusters to remove
        discard=discard+1;
    end
end
disp(strcat(['Identified ' num2str(nclusters-discard) ' valid
nanoparticles from ' num2str(nclusters) ' candidate
nanoparticles']));
NPselect= R~=0;
R=R(NPselect);
R=round(R);
C=[C(NPselect,1), C(NPselect,2)];
NPSize647 = NPSize647(IndexToRemove);  %Pietro: remove the discarded
cluster, after size check
NPMembs=NPMembs(IndexToRemove); %Pietro: remove discarded clusters,
after size check

% Pietro: filling the Loc2particle cell array with selected X,Y,T
coords for each particle:
figure(2)
title({'black ellipse: circle fitting';'red dots: selected-NP
localiz, black dots: other localiz'});grid on;axis equal;
hold on
plot(XCoords647,YCoords647,'.k'); axis equal; hold on;

Lgt=length(NPMembs);
Loc2particle = cell(Lgt,1); %initialize cell array, number of
entries corresponding to numb of selected NPs

%fill the cell array: for each selected NP the X,Y,T coords of
localizations
for k = 1 : Lgt
    A = [XCoords647(cell2mat(NPMembs(k)))';
YCoords647(cell2mat(NPMembs(k)))'; TCoords647(cell2mat(NPMembs(k)))'
];
    Loc2particle{k}=A;
    plot(A(1,:), A(2,:), 'r.'); hold on; %plot selected localiz in
red
end

%this is to plot the retrieved radius for selected NPs:
for m=1:length(R)
    t = linspace(0,2*pi,100);
    plot(R(m)*cos(t)+C(m,1),R(m)*sin(t)+C(m,2),'k','LineWidth', 1)
    hold on
%   axis image
    %axis([C(m,1)-500 C(m,1)+500 C(m,2)-500 C(m,2)+500])
end

% Statistics and export:
```

```
ClustSize=NPSize647;
% ClustSize = cellfun(@numel, Loc2particle); %how many
647Localizations are in each nanoparticle
% ClustSize=ClustSize/3; %IMPORTANT! Because count 1 localization as
3 (3 coords)

%histograms of local and diameter:
figure(3);hist(ClustSize,30);grid on;title('Number of 647
localizations/nanoparticle');
figure(4);hist(R*2,10);grid on;title('Nanoparticle diam');

% %Pietro: export some useful result
 diam=R*2;
% save NPloc.out ClustSize -ascii
% save NPdiam.out diam -ascii




% ClusSize=Bandwidth;
% diam=MaxDiam;
% Loc2particle=XCoords647;

end
```

11.2.1.   Auxiliary functions and methods

Fit_ellipse.m

```
function ellipse_t =
fit_ellipse(x,y,axis_handle,ScaleFactor,MaxParticleElongation,MaxPar
ticleDiameter,isolated)
%
% fit_ellipse - finds the best fit to an ellipse for the given set
of points.
%
% Format:   ellipse_t = fit_ellipse( x,y,axis_handle )
%
% Input:    x,y         - a set of points in 2 column vectors. AT
LEAST 5 points are needed !
%           axis_handle - optional. a handle to an axis, at which
the estimated ellipse
%                         will be drawn along with it's axes
%
% Output:   ellipse_t - structure that defines the best fit to an
ellipse
%                       a          - sub axis (radius) of the X
axis of the non-tilt ellipse
%                       b          - sub axis (radius) of the Y
axis of the non-tilt ellipse
%                       phi        - orientation in radians of the
ellipse (tilt)
%                       X0         - center at the X axis of the
non-tilt ellipse
%                       Y0         - center at the Y axis of the
non-tilt ellipse
%                       X0_in      - center at the X axis of the
tilted ellipse
%                       Y0_in      - center at the Y axis of the
tilted ellipse
```

```
%                              long_axis    - size of the long axis of the
ellipse
%                              short_axis   - size of the short axis of the
ellipse
%                              status       - status of detection of an
ellipse
%
% Note:      if an ellipse was not detected (but a parabola or
hyperbola), then
%            an empty structure is returned


%
=====================================================================
================
%                    Ellipse Fit using Least Squares criterion
%
=====================================================================
================
% We will try to fit the best ellipse to the given measurements. the
mathematical
% representation of use will be the CONIC Equation of the Ellipse
which is:
%
%    Ellipse = a*x^2 + b*x*y + c*y^2 + d*x + e*y + f = 0
%
% The fit-estimation method of use is the Least Squares method
(without any weights)
% The estimator is extracted from the following equations:
%
%    g(x,y;A) := a*x^2 + b*x*y + c*y^2 + d*x + e*y = f
%
%    where:
%        A   - is the vector of parameters to be estimated
(a,b,c,d,e)
%        x,y - is a single measurement
%
% We will define the cost function to be:
%
%   Cost(A) := (g_c(x_c,y_c;A)-f_c)'*(g_c(x_c,y_c;A)-f_c)
%            = (X*A+f_c)'*(X*A+f_c)
%            = A'*X'*X*A + 2*f_c'*X*A + N*f^2
%
%    where:
%       g_c(x_c,y_c;A) - vector function of ALL the measurements
%                        each element of g_c() is g(x,y;A)
%        X             - a matrix of the form: [x_c.^2, x_c.*y_c,
y_c.^2, x_c, y_c ]
%        f_c           - is actually defined as ones(length(f),1)*f
%
% Derivation of the Cost function with respect to the vector of
parameters "A" yields:
%
%   A'*X'*X = -f_c'*X = -f*ones(1,length(f_c))*X = -f*sum(X)
%
% Which yields the estimator:
%
%
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~~~~~~~~
%        |  A_least_squares = -f*sum(X)/(X'*X) ->(normalize by -f) =
sum(X)/(X'*X)   |
```

```
%
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~~~~~~~~~
%
% (We will normalize the variables by (-f) since "f" is unknown and
can be accounted for later on)
%
% NOW, all that is left to do is to extract the parameters from the
Conic Equation.
% We will deal the vector A into the variables: (A,B,C,D,E) and
assume F = -1;
%
%    Recall the conic representation of an ellipse:
%
%       A*x^2 + B*x*y + C*y^2 + D*x + E*y + F = 0
%
% We will check if the ellipse has a tilt (=orientation). The
orientation is present
% if the coefficient of the term "x*y" is not zero. If so, we first
need to remove the
% tilt of the ellipse.
%
% If the parameter "B" is not equal to zero, then we have an
orientation (tilt) to the ellipse.
% we will remove the tilt of the ellipse so as to remain with a
conic representation of an
% ellipse without a tilt, for which the math is more simple:
%
% Non tilt conic rep.:  A`*x^2 + C`*y^2 + D`*x + E`*y + F` = 0
%
% We will remove the orientation using the following substitution:
%
%    Replace x with cx+sy and y with -sx+cy such that the conic
representation is:
%
%    A(cx+sy)^2 + B(cx+sy)(-sx+cy) + C(-sx+cy)^2 + D(cx+sy) + E(-
sx+cy) + F = 0
%
%    where:      c = cos(phi)    ,    s = sin(phi)
%
%    and simplify...
%
%       x^2(A*c^2 - Bcs + Cs^2) + xy(2A*cs +(c^2-s^2)B -2Ccs) + ...
%          y^2(As^2 + Bcs + Cc^2) + x(Dc-Es) + y(Ds+Ec) + F = 0
%
%    The orientation is easily found by the condition of (B_new=0)
which results in:
%
%    2A*cs +(c^2-s^2)B -2Ccs = 0  ==> phi = 1/2 * atan( b/(c-a) )
%
%    Now the constants   c=cos(phi)  and  s=sin(phi)  can be found,
and from them
%    all the other constants A`,C`,D`,E` can be found.
%
%    A` = A*c^2 - B*c*s + C*s^2                    D` = D*c-E*s
%    B` = 2*A*c*s +(c^2-s^2)*B -2*C*c*s = 0        E` = D*s+E*c
%    C` = A*s^2 + B*c*s + C*c^2
%
% Next, we want the representation of the non-tilted ellipse to be
as:
%
```

```matlab
%         Ellipse = ( (X-X0)/a )^2 + ( (Y-Y0)/b )^2 = 1
%
%         where:  (X0,Y0) is the center of the ellipse
%                 a,b     are the ellipse "radiuses" (or sub-axis)
%
% Using a square completion method we will define:
%
%         F`` = -F` + (D`^2)/(4*A`) + (E`^2)/(4*C`)
%
%         Such that:    a`*(X-X0)^2 = A`(X^2 + X*D`/A` + (D`/(2*A`))^2
)
%                       c`*(Y-Y0)^2 = C`(Y^2 + Y*E`/C` + (E`/(2*C`))^2
)
%
%         which yields the transformations:
%
%             X0  =   -D`/(2*A`)
%             Y0  =   -E`/(2*C`)
%             a   =   sqrt( abs( F``/A` ) )
%             b   =   sqrt( abs( F``/C` ) )
%
% And finally we can define the remaining parameters:
%
%   long_axis   = 2 * max( a,b )
%   short_axis  = 2 * min( a,b )
%   Orientation = phi
%
%

% initialize
orientation_tolerance = 1e-3;

% empty warning stack
warning( '' );

% prepare vectors, must be column vectors
x = x(:);
y = y(:);

% remove bias of the ellipse - to make matrix inversion more
accurate. (will be added later on).
mean_x = mean(x);
mean_y = mean(y);
x = x-mean_x;
y = y-mean_y;

% the estimation for the conic equation of the ellipse
X = [x.^2, x.*y, y.^2, x, y ];
a = sum(X)/(X'*X);

% check for warnings
if ~isempty( lastwarn )
    disp( 'stopped because of a warning regarding matrix inversion'
);
    ellipse_t = [];
    return
end

% extract parameters from the conic equation
[a,b,c,d,e] = deal( a(1),a(2),a(3),a(4),a(5) );
```

```matlab
% remove the orientation from the ellipse
if ( min(abs(b/a),abs(b/c)) > orientation_tolerance )

    orientation_rad = 1/2 * atan( b/(c-a) );
    cos_phi = cos( orientation_rad );
    sin_phi = sin( orientation_rad );
    [a,b,c,d,e] = deal(...
        a*cos_phi^2 - b*cos_phi*sin_phi + c*sin_phi^2,...
        0,...
        a*sin_phi^2 + b*cos_phi*sin_phi + c*cos_phi^2,...
        d*cos_phi - e*sin_phi,...
        d*sin_phi + e*cos_phi );
    [mean_x,mean_y] = deal( ...
        cos_phi*mean_x - sin_phi*mean_y,...
        sin_phi*mean_x + cos_phi*mean_y );
else
    orientation_rad = 0;
    cos_phi = cos( orientation_rad );
    sin_phi = sin( orientation_rad );
end

% check if conic equation represents an ellipse
test = a*c;
switch (1)
case (test>0),  status = '';
case (test==0), status = 'Parabola found';  warning( 'fit_ellipse:
Did not locate an ellipse' );
case (test<0),  status = 'Hyperbola found'; warning( 'fit_ellipse:
Did not locate an ellipse' );
end

% if we found an ellipse return it's data
if (test>0)

    % make sure coefficients are positive as required
    if (a<0), [a,c,d,e] = deal( -a,-c,-d,-e ); end

    % final ellipse parameters
    X0          = mean_x - d/2/a;
    Y0          = mean_y - e/2/c;
    F           = 1 + (d^2)/(4*a) + (e^2)/(4*c);
    [a,b]       = deal( sqrt( F/a ),sqrt( F/c ) );
    long_axis   = 2*max(a,b);
    short_axis  = 2*min(a,b);

    % rotate the axes backwards to find the center point of the
original TILTED ellipse
    R           = [ cos_phi sin_phi; -sin_phi cos_phi ];
    P_in        = R * [X0;Y0];
    X0_in       = P_in(1);
    Y0_in       = P_in(2);

    % pack ellipse into a structure
    ellipse_t = struct( ...
        'a',a,...
        'b',b,...
        'phi',orientation_rad,...
        'X0',X0,...
        'Y0',Y0,...
        'X0_in',X0_in,...
        'Y0_in',Y0_in,...
```

```matlab
        'long_axis',long_axis,...
        'short_axis',short_axis,...
        'status','' );
else
    % report an empty structure
    ellipse_t = struct( ...
        'a',[],...
        'b',[],...
        'phi',[],...
        'X0',[],...
        'Y0',[],...
        'X0_in',[],...
        'Y0_in',[],...
        'long_axis',[],...
        'short_axis',[],...
        'status',status );
end

if (test<0)                              %added by natalia: if
test<0 then in the loop crashes because it doesn't have the
parameter ellipse_t.valid
    ellipse_t.valid = 0;
end

% check if we need to plot an ellipse with it's axes.
if (nargin>2) & ~isempty( axis_handle ) & (test>0)

    a = a*ScaleFactor;
    b = b*ScaleFactor;
    ellipse_t.long_axis = ellipse_t.long_axis*ScaleFactor;
    ellipse_t.short_axis = ellipse_t.short_axis*ScaleFactor;
    elongation = ellipse_t.long_axis/ellipse_t.short_axis;
    %disp([elongation ellipse_t.long_axis ellipse_t.short_axis]);
    if (elongation <= MaxParticleElongation) && (ellipse_t.long_axis
<= MaxParticleDiameter)
        ellipse_t.valid = 1;
    else
        ellipse_t.valid = 0;
      % elongation
      % ellipse_t.long_axis
    end

    % rotation matrix to rotate the axes with respect to an angle
phi
    R = [ cos_phi sin_phi; -sin_phi cos_phi ];

    % the axes
    ver_line        = [ [X0 X0]; Y0+b*[-1 1] ];
    horz_line       = [ X0+a*[-1 1]; [Y0 Y0] ];
    new_ver_line    = R*ver_line;
    new_horz_line   = R*horz_line;

    % the ellipse
    theta_r         = linspace(0,2*pi);
    ellipse_x_r     = X0 + a*cos( theta_r );
    ellipse_y_r     = Y0 + b*sin( theta_r );
    rotated_ellipse = R * [ellipse_x_r;ellipse_y_r];

    % draw
    hold_state = get( axis_handle,'NextPlot' );
    set( axis_handle,'NextPlot','add' );
```

```
    if isolated
        if ellipse_t.valid == 1
            color = 'om';
        else
            color = 'oc';
        end
    else
       color = 'oy';
        ellipse_t.valid = 0;
    end

    plot( new_ver_line(1,:),new_ver_line(2,:),color,'LineWidth', 1);
    plot( new_horz_line(1,:),new_horz_line(2,:),color,'LineWidth',
1);
    plot(
rotated_ellipse(1,:),rotated_ellipse(2,:),color,'LineWidth', 1);
    set( axis_handle,'NextPlot',hold_state );
end
```

MeanShiftCluster.m

```
function [clustCent,data2cluster,cluster2dataCell] =
MeanShiftCluster(dataPts,bandWidth,plotFlag);
%perform MeanShift Clustering of data using a flat kernel
%
% ---INPUT---
% dataPts           - input data, (numDim x numPts)
% bandWidth         - is bandwidth parameter (scalar)
% plotFlag          - display output if 2 or 3 D    (logical)
% ---OUTPUT---
% clustCent         - is locations of cluster centers (numDim x
numClust)
% data2cluster      - for every data point which cluster it belongs
to (numPts)
% cluster2dataCell  - for every cluster which points are in it
(numClust)
%
% Bryan Feldman 02/24/06
% MeanShift first appears in
% K. Funkunaga and L.D. Hosteler, "The Estimation of the Gradient of
a
% Density Function, with Applications in Pattern Recognition"


%*** Check input ****
if nargin < 2
    error('no bandwidth specified')
end

if nargin < 3
    plotFlag = true;
    plotFlag = false;
end

%**** Initialize stuff ***
[numDim,numPts] = size(dataPts);
numClust        = 0;
bandSq          = bandWidth^2;
initPtInds      = 1:numPts;
```

```matlab
maxPos          = max(dataPts,[],2);
%biggest size in each dimension
minPos          = min(dataPts,[],2);
%smallest size in each dimension
boundBox        = maxPos-minPos;                        %bounding
box size
sizeSpace       = norm(boundBox);                       %indicator
of size of data space
stopThresh      = 1e-3*bandWidth;                       %when mean
has converged
clustCent       = [];                                   %center of
clust
beenVisitedFlag = zeros(1,numPts,'uint8');              %track if a
points been seen already
numInitPts      = numPts;                               %number of
points to posibaly use as initilization points
clusterVotes    = zeros(1,numPts,'uint16');             %used to
resolve conflicts on cluster membership


while numInitPts

    tempInd         = ceil( (numInitPts-1e-6)*rand);        %pick a
random seed point /ceil=redondeo hacia + infinito, redondea a numero
entero
    stInd           = initPtInds(tempInd);                  %use
this point as start of mean
    myMean          = dataPts(:,stInd);                             %
intilize mean to this points location
    myMembers       = [];                                   % points
that will get added to this cluster
    thisClusterVotes    = zeros(1,numPts,'uint16');         %used to
resolve conflicts on cluster membership

    while 1      %loop untill convergence

        sqDistToAll = sum((repmat(myMean,1,numPts) - dataPts).^2);
%dist squared from mean to all points still active  Natalia:
repmat(A,M,N) creates a large matrix B consisting of an M-by-N
tiling of copies of A
        inInds      = find(sqDistToAll < bandSq);
%points within bandWidth
        thisClusterVotes(inInds) = thisClusterVotes(inInds)+1;  %add
a vote for all the in points belonging to this cluster


        myOldMean   = myMean;
%save the old mean
        myMean      = mean(dataPts(:,inInds),2);
%compute the new mean
        myMembers   = [myMembers inInds];                      %add
any point within bandWidth to the cluster
        beenVisitedFlag(myMembers) = 1;
%mark that these points have been visited

        %*** plot stuff ****
        if plotFlag
            figure(12345),clf,hold on
            if numDim == 2
                plot(dataPts(1,:),dataPts(2,:),'.')
                plot(dataPts(1,myMembers),dataPts(2,myMembers),'ys')
```

```matlab
                plot(myMean(1),myMean(2),'go')
                plot(myOldMean(1),myOldMean(2),'rd')
                pause
            end
        end

        %**** if mean doesn't move much stop this cluster ***
        if norm(myMean-myOldMean) < stopThresh

            %check for merge posibilities
            mergeWith = 0;
            for cN = 1:numClust
                distToOther = norm(myMean-clustCent(:,cN));
%distance from posible new clust max to old clust max
                if distToOther < bandWidth/2                    %if
its within bandwidth/2 merge new and old
                    mergeWith = cN;
                    break;
                end
            end


            if mergeWith > 0    % something to merge
                clustCent(:,mergeWith)       =
0.5*(myMean+clustCent(:,mergeWith));            %record the max as
the mean of the two merged (I know biased twoards new ones)
                %clustMembsCell{mergeWith}    =
unique([clustMembsCell{mergeWith} myMembers]);   %record which
points inside
                clusterVotes(mergeWith,:)    =
clusterVotes(mergeWith,:) + thisClusterVotes;    %add these votes to
the merged cluster
            else    %its a new cluster
                numClust                 = numClust+1;
%increment clusters
                clustCent(:,numClust)        = myMean;
%record the mean
                %clustMembsCell{numClust}    = myMembers;
%store my members
                clusterVotes(numClust,:)    = thisClusterVotes;
            end

            break;
        end

    end


    initPtInds      = find(beenVisitedFlag == 0);           %we can
initialize with any of the points not yet visited
    numInitPts      = length(initPtInds);                  %number
of active points in set

end

[val,data2cluster] = max(clusterVotes,[],1);                %a point
belongs to the cluster with the most votes

%*** If they want the cluster2data cell find it for them
if nargout > 2
    cluster2dataCell = cell(numClust,1);
```

```
    for cN = 1:numClust
        myMembers = find(data2cluster == cN);
        cluster2dataCell{cN} = myMembers;
    end
end
```

11.3.   Other graphs

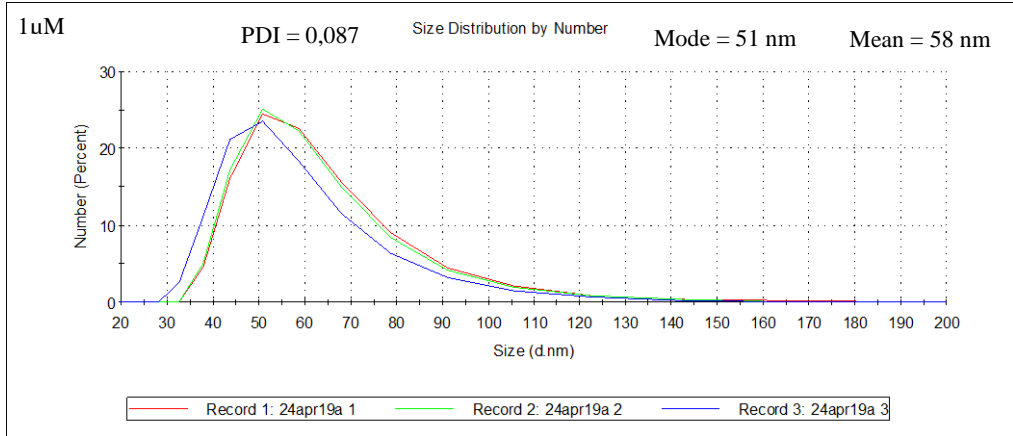### 11.3.1.   Size distribution analysis between different dye concentrations
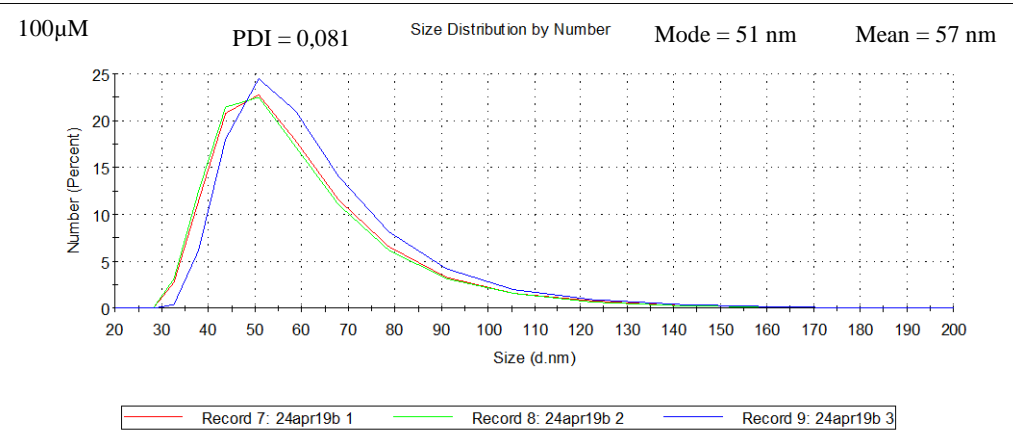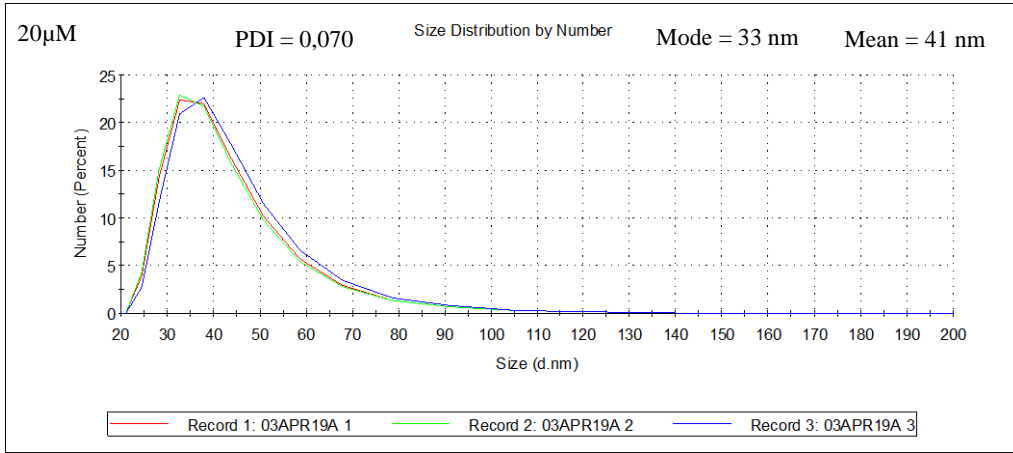
Figure 11. Size distribution analysis by DLS between samples with different concentration of the dye DiD: 1 µM, 20 µM and 100 µM.
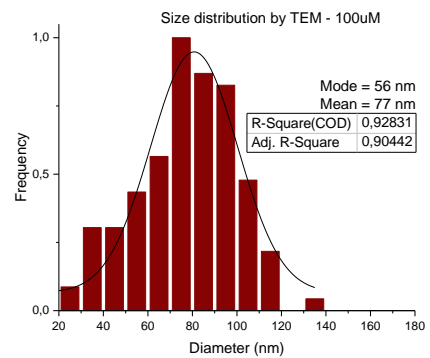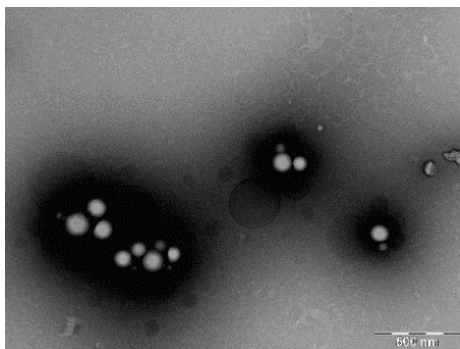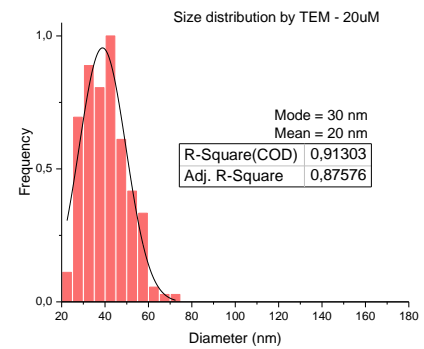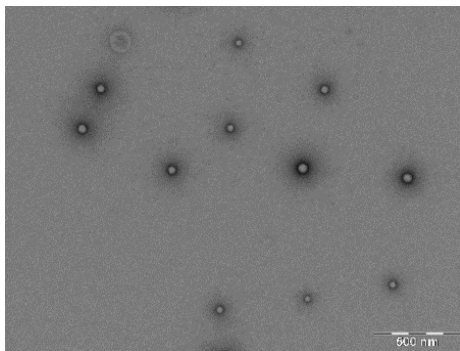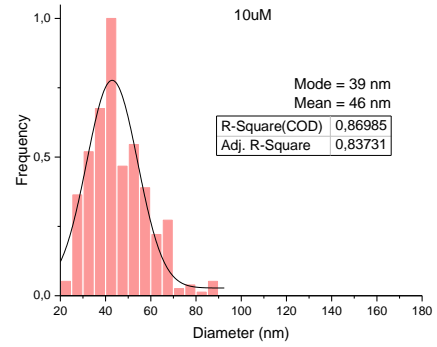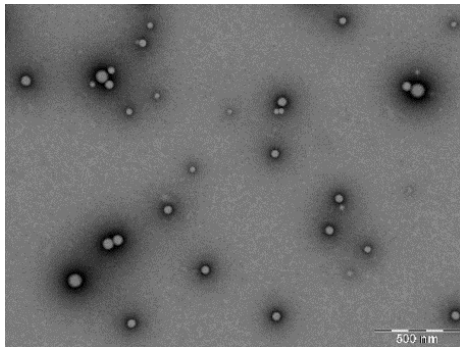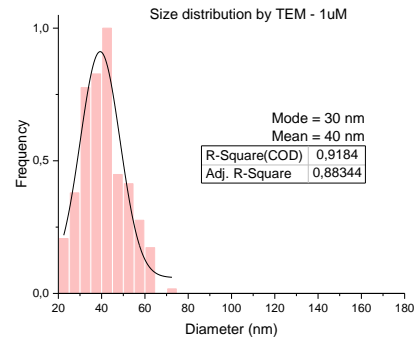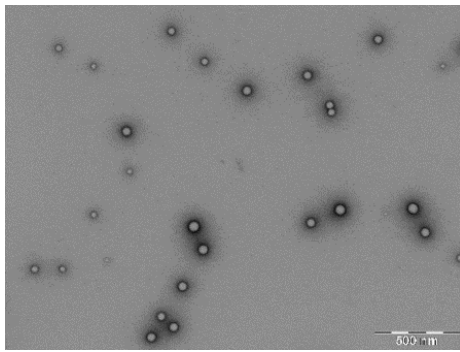
Figure 12. Size distribution analysis by TEM between samples with different concentration of the dye DiD: 1 µM, 10 µM, 20 µM and 100 µM.
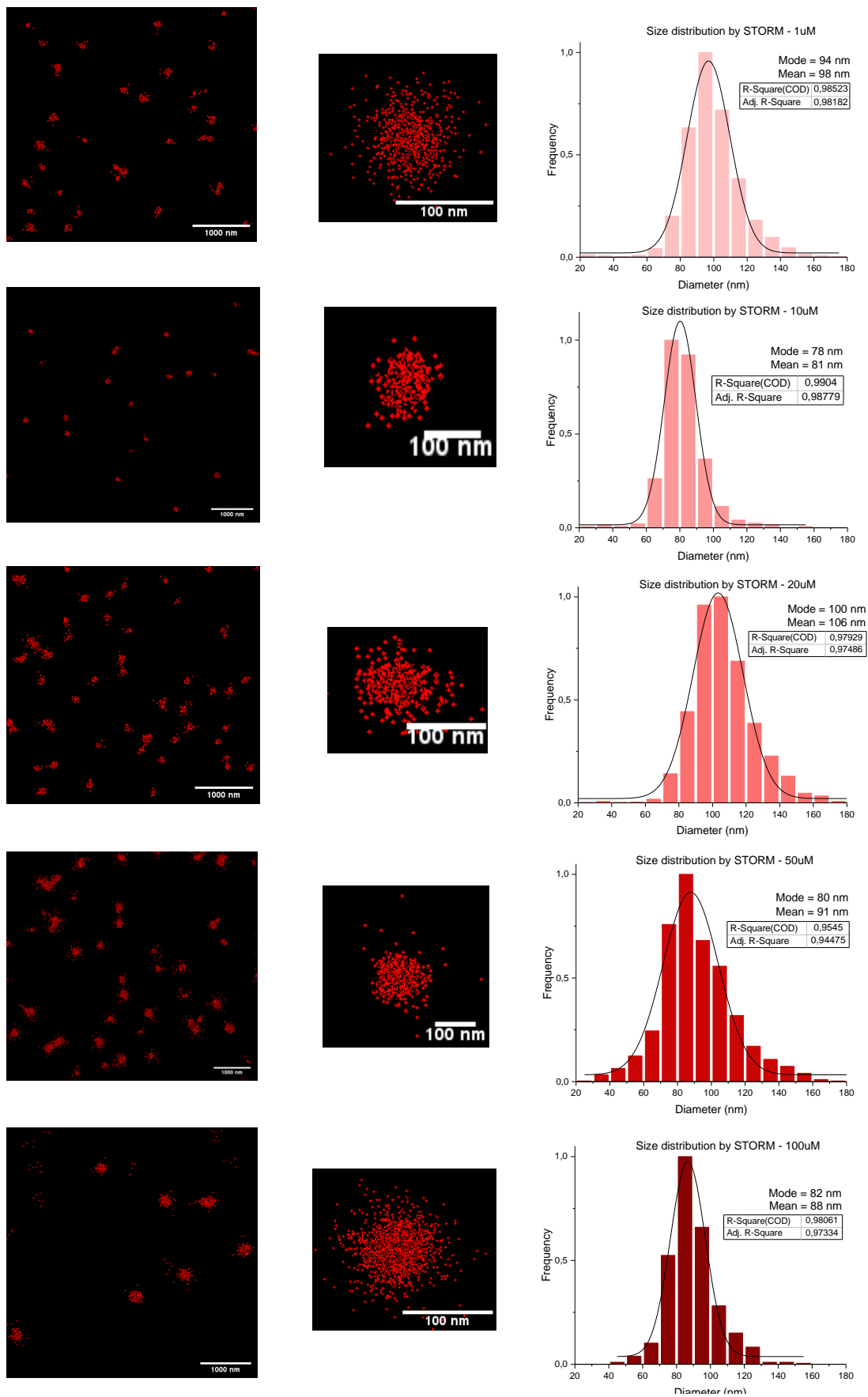
Figure 13. Size distribution analysis by STORM between samples with different concentration of the dye DiD: 1 µM, 10 µM, 20 µM, 50 µM and 100 µM.

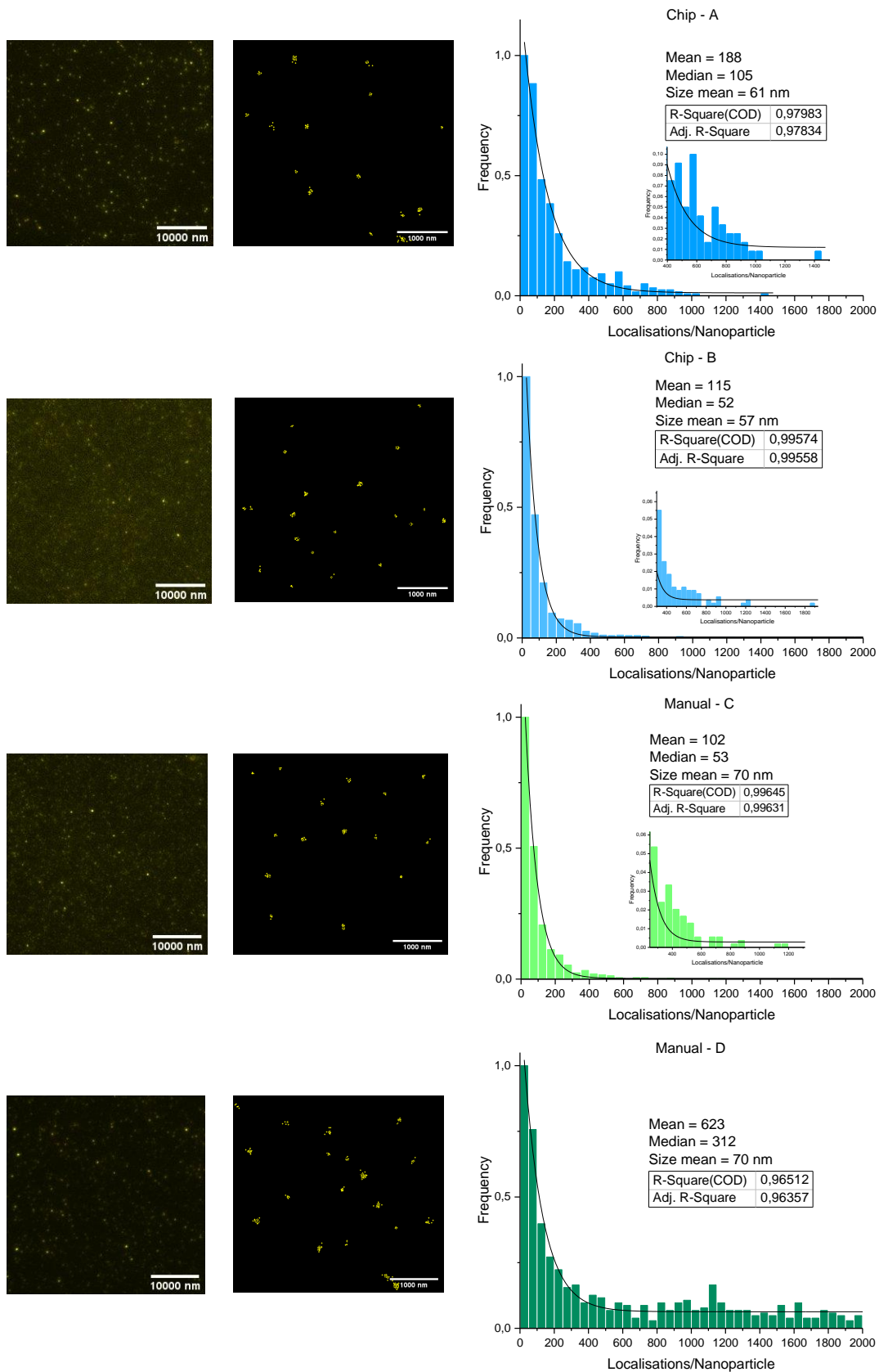## 11.3.2. Encapsulation efficiency between different formulation methods

Figure 14. Encapsulation efficiency by STORM between different formulations (microfluidic device and manual nanoprecipitation).

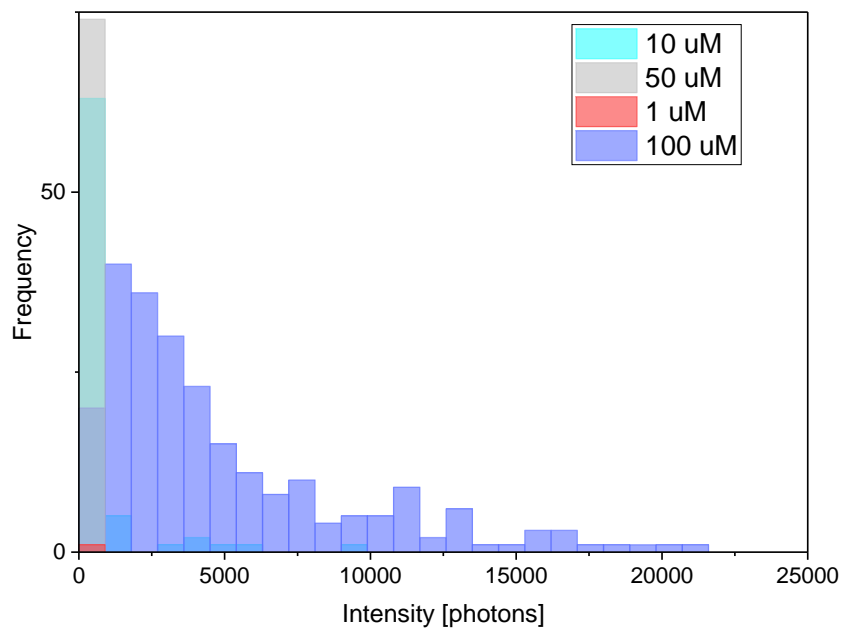### 11.3.3.   Encapsulation efficiency by ThunderSTORM

Figure 15. Encapsulation efficiency by the ImageJ plugin ThunderSTORM between samples with different concentration of the dye DiD: 1 µM, 10 µM, 50 µM and 100 µM. Provided by PhD fellow Adrianna Glinkowska.