



Model reconstruction of fiber yarn from CT images

Josef Vosáhlo¹ 

¹Technical University of Liberec, Studentska 2, 461 17 Liberec 1, Czech Republic
josef.vosahlo@tul.cz

Abstract. The Image analysis of fibers is a key technique in the field of materials research and industrial engineering. The purpose of this work is to present a solution for fiber image analysis using MATLAB, a high-performance software environment for numerical computations and data visualization. This work focused on the detection, segmentation, and quantification of fibers in microscopic images. Initially, image preprocessing techniques such as filtering and adaptive thresholding were applied to enhance contrast and reduce noise. Subsequent steps utilized edge detection algorithms, like the Canny detector, and fiber segmentation, for instance using Watershed transformation. Quantification of the fibers was carried out by extracting geometric and textural features, such as length, diameter, orientation, and energy. These properties provide important information about the structure and characteristics of the fibers. The results of the implementation show high accuracy in detecting and segmenting the central points of fibers and provide important information for further material analysis. This work presents an efficient and robust solution for fiber image analysis using MATLAB, which can be applied in various areas, such as the textile industry, biomedical research, or materials engineering. This approach opens new possibilities for the automation of fiber analysis and accelerates the processes of research and development of materials. The created work contributes to a better understanding of the structure and properties of fibers and enhances the potential of their use in a wide range of applications.

Keywords: CT images, Matlab, 3D geometry, script

1 Introduction

In the field of material research and industrial engineering, fiber image analysis plays a crucial role in the study and characterization of materials. With growing interest in advanced image processing methods, it becomes important to develop efficient and reliable procedures for the detection, segmentation, and quantification of fibers. In this context, MATLAB, a powerful software environment for numerical computations and data visualization, offers broad possibilities for solving fiber image analysis. The goal of this work is to present a comprehensive solution for fiber image analysis using MATLAB. The work was focused on the use of advanced algorithms for the detection, segmentation, and quantification of fibers in microscopic images of materials. The introductory part of the work focuses on image preprocessing, which includes

© The Author(s) 2024

M. Petr et al. (eds.), *62nd International Conference of Machine Design Departments (ICMD 2022)*,

Atlantis Highlights in Engineering 28,

https://doi.org/10.2991/978-94-6463-423-5_13

techniques of filtering, adaptive thresholding, and contrast enhancement. The next step is the application of algorithms for edge detection and fiber segmentation. We use modern methods such as the Canny detector and Watershed transformation, which allow for precise identification and separation of individual fibers. The subsequent step involved the application of algorithms for the detection of central points of the already segmented fibers. For subsequent work, the created central points were exported to an Excel file using an algorithm. For the quantification of fibers, we then extract geometric and textural properties, such as length, diameter, and orientation. These properties provide important information about the structure and characteristics of the fibers, which can be used for further analysis and applications. The results of the created implementation show high accuracy and reliability of fiber detection and segmentation, which represents a significant contribution to the field of material research and industrial engineering. The work created provides an efficient and robust solution for fiber image analysis using MATLAB, which can be applied in various sectors, such as the textile industry, biomedical research, and material engineering [1,2,3].

2 Materials and Methods

2.1 Computed tomography

Computed tomography (CT) is an imaging method that allows for the construction of a 3D model of the internal and external structure of an examined object based on X-ray images taken from different angles, without the need to damage the object. This makes CT suitable for detailed examination of macro objects [3]. In conventional radiography, the subject is placed between the source of X-ray radiation and a sensitive X-ray film. As the radiation passes through the object, it is absorbed depending on the material or combination of different materials the object is composed of, more precisely on their absorption coefficients. By detecting the intensity of radiation behind the object, a radiograph is obtained, which is an image of varying shades of gray depending on the attenuation of radiation in the object. Replacing X-ray film with a radiation detector allows for the digitization of the image and its subsequent processing [4]. The principle of computed tomography is to obtain a set of images of the same object taken with angular increments. From the obtained set of images, a 3D model of the examined object, known as reconstruction, is calculated using a mathematical algorithm [4]. The basic setup of a laboratory tomography consists of an X-ray radiation source, an X-ray radiation detector, and a positioning system. Standard CT equipment used in medicine, for example, is based on the principle of a statically placed object around which the radiation source and detector rotate. In the scientific sphere, the opposite principle is often used, where the radiation source and detector are static and the object is rotated, as shown in Fig. 1 (a,b).

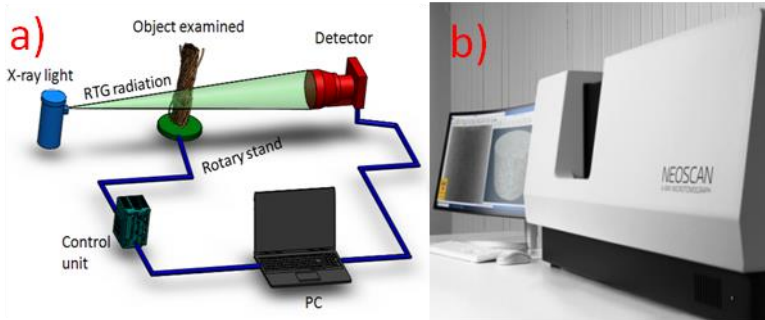


Fig. 1 Schematic of laboratory CT, b) Scanning CT equipment from the company Neoscan.

Advantages

- Enables visualization of low-contrast parts of individual fibers.
- High sensitivity of scintillation detectors.
- Very rapid data processing from the scintillation detector.
- Very good resolution and contrast, meaning sharper images.
- Elimination of reflections and their interference.
- Results can be stored or transferred to cloud storage.
- Selected sections of images can be examined in more detail.

Disadvantages

- High acquisition cost of the equipment.
- Operation of CT requires a high level of trained personnel.

2.2 Image tomography analysis

The CT images created were saved in a gray scale bitmap format *.tif. The basic display of these images allowed for a view of the cross-section of an 800 μm fiber yarn with a thickness of a single layer of 1 μm . In the initial view of the image, it was evident that most of the fibers in the fiber system are in close contact and a reconstructed 3D model could not be created from these images. See Fig. 2.

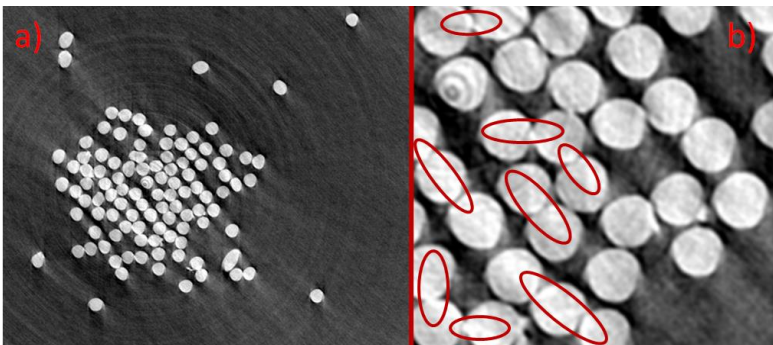


Fig. 2 a) Basic view of a single-section CT image. b) Detection of fibers in contact.

Based on the detected fibers, adjustments to CT images can be made to separate the fiber contacts from each other. To achieve the separation of individual fibers, a script was created in Matlab R2021a that divides the fibers into individual segments, thus facilitating the creation of separate geometries for the isolated fibers. Before beginning the process, it was necessary to rename all the images so that the script could sequentially load the uniform images. In the subsequent step, a block diagram of the script was created, which was followed throughout the script creation process. See Fig. 2.2 [3, 4].

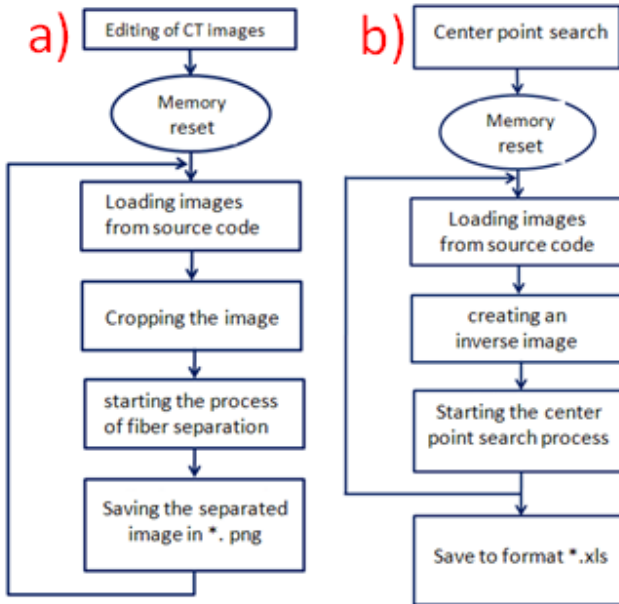


Fig. 3 a) Block diagram of separated fibers b) Block diagram of finding the center point.

3 Results and Discussion

3.1 Creating a thread separation script

Matlab is designed for multifunctional use. It is used not only for processing measured data but also for processing data from image analysis. To create a script that would separate merged fibers, it was necessary to create binary images from the CT scans, see Fig. 5 a). In the first step, a function for the convergence of binary images was created, which was used in the *Colorthresholder* application. This application allowed for the adjustment of saturation and brightness parameters, see Fig 4 (a, b). The modified image was then converted into a binary black and white spectrum using the *Show Binary* function [5]. The newly created image could be saved to temporary memory for further processing. see Fig. 4 a)

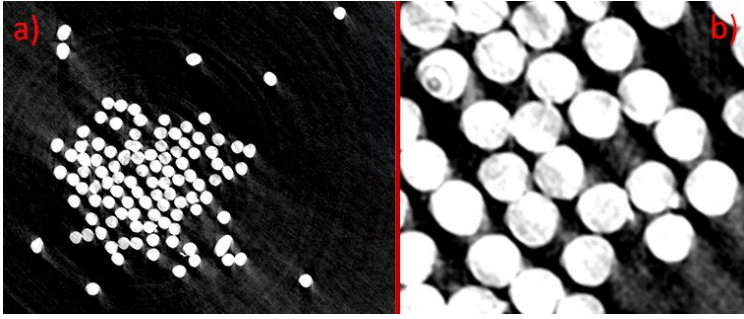


Fig. 4 a) CT image with increased saturation and light, b) Detailed view CT image of fibers

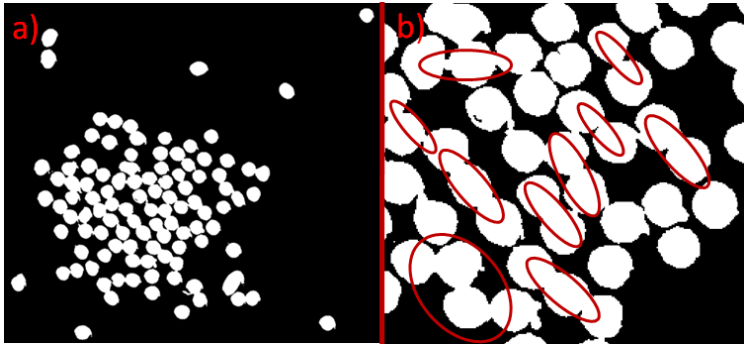


Fig. 5 a) Converged binary CT image b) Detailed view of the binary image with intersections

The stored function for processing the binary image was utilized for all existing images. An application named *Image Batch Processor* was used for the purpose of duplicating the function, which allowed repeating the stored process in Matlab memory arbitrarily and subsequently saving the created data into a chosen file. To verify the theory of yarn construction from individual CT slices, the first 100 CT images out of a total of 2500 images were used. The processed results of the binary images visibly indicate the problem of the interpenetration of individual fibers, see the detailed view in Fig. 5 b). To remove the interpenetration of individual fibers, the stored binary images were further used in the *Color Threshold* application. As in previous steps, the function for copying the properties of the *Image batch processor* for 100 basic images was utilized. To load the modified binary images, it is necessary to enter the full path to the root directory where the individual images will be saved. For this specific task of separating fibers in interpenetration, the method of water transformation was chosen. The selected function is tasked with finding and marking the most overlapping fibers in the given object. In the image Fig (a, b), it is possible to see the loaded binary fibers and the already generated so-called water transformation, which marked the most overlapping fibers in the binary image. The newly formed shapes highlight the most affected areas where the fibers interpenetrated, which are displayed in Fig. 6 b). To display the so-called water transformation, commands were used, see equation 1 [6].

$$\left. \begin{array}{l} bw1 = imread(bw); \\ K = watershed(bw1); \\ Lrgb = label2rgb(K); \\ imshow(Lrgb) \end{array} \right\} \quad (1)$$

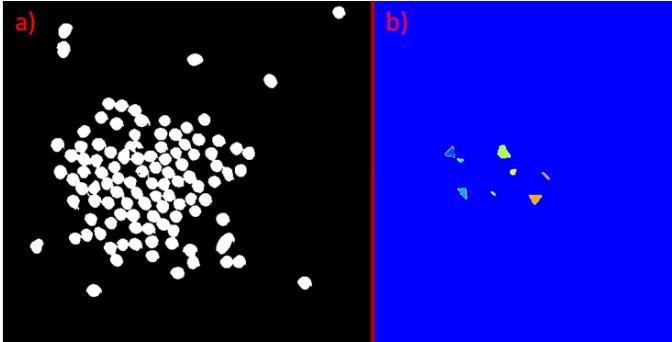


Fig. 6 a) Binary display of fibers, b) Display of the so-called water transformation

In the first generated preview, the image 6 b) shows a blue background, the so-called water surface, in which shapes generated by the so-called water transformation are found. For a more detailed display, a preview of the transformed places that the program selected for further segregation resolution can be seen in Fig. 7. The script function, see equation 2, overlays the basic binary image with the water transformation. In Fig. 7 (a, b), the fibers colored green, which the program used to separate the most affected areas of fiber interpenetration, are visible. Commands were used to display the separation, see equation 2 [7].

$$\left. \begin{array}{l} imshow(imfuse(bw1, Lrgb)); \\ axis([10 175 15 155]); \end{array} \right\} \quad (2)$$

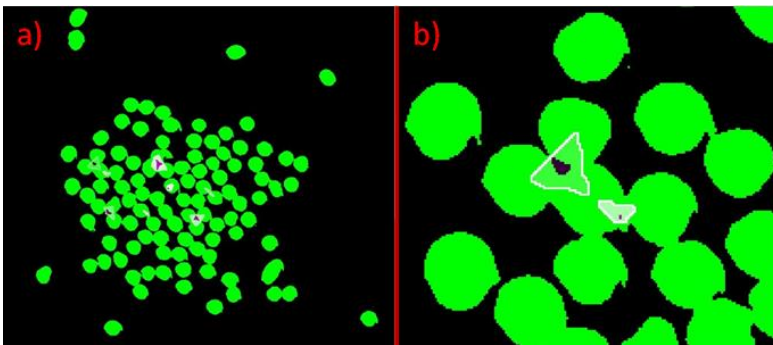


Fig. 7 a) Binary image with the water transformation, b) Detailed transformation

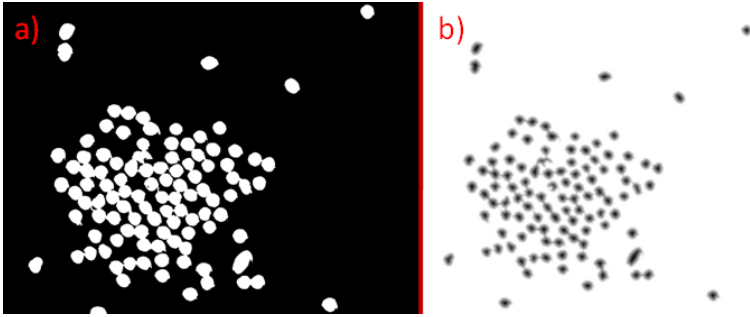


Fig. 8 a) Binary representation of fiber cross-section, b) Denoising and representation of *bwareaopen* function

The next step was the creation of identification for fiber segments that intersect with another segment. The distance transformation can be set with associated displayed objects that will be identified. Before starting the distance transformation, it is necessary to perform noise cleaning in the image [8]. The function *bwareaopen* can be used to remove very small points. This function removes only foreground impurities, so we supplement the image Fig. 8 a) before and after invoking the command *bwareaopen*, see Fig. 8 b). Commands for displaying the so-called *bwareaopen* were used, see equation 3.

$$\left. \begin{aligned} bw2 &= \sim bwareaopen(\sim bw1, 10); \\ imshow(bw2) \\ D &= -bwdist(\sim bw); \\ imshow(D, []) \end{aligned} \right\} \quad (3)$$

$$\left. \begin{aligned} Ld &= watershed(D); \\ imshow(label2rgb(Ld)) \end{aligned} \right\} \quad (4)$$

The next step, which follows from equation 4, where the repeat command (*watershed*) was used again. The so-called water transformation divided all objects fibers into segments as shown in Fig. 9 a) and the detailed view 9 b).

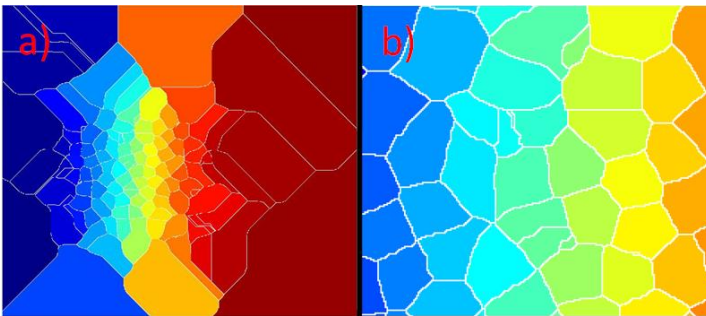


Fig. 9 a) New water transformation, b) Detailed display of water transformation.

The created boundary lines of the segments as shown in Fig. 9 b) correspond to the division of individual fibers, meaning that in white color $Ld == 0$. If we use these boundary lines to segment the binary image, they will correspond to the respective changes of pixels located on the background of the binary image, which follows from equation 5.

$$\left. \begin{array}{l} bw2 = bw1; \\ bw2(Ld == 0) = 0; \\ imshow(bw2) \end{array} \right\} \quad (5)$$

The so-called "raw" watershed transformation is known for its tendency to "over-segment" an image. A common practice in segmentation methods based on the *watershed* function is to filter out small local minima using *imextendedmin* and then to modify the distance transformation so that no minima occur at the filtered locations. This is called "minima imposition" and is implemented using the function *imimposemin*. By the following command, I invoke the function *imextendedmin*, which should ideally create only small spots that are roughly in the center of the cells that are to be further segmented. With the subsequent use of the function *imshowpair*, an overlay of the mask on the original image was created, as shown in equation 6.

$$\left. \begin{array}{l} mask = imextendedmin(D, 2); \\ imshowpair(bw, mask, 'blend') \end{array} \right\} \quad (6)$$

Finally, the transformation function allowed for the adjustment of individual distances so that the minima were contained only at the desired locations, and then the (watershed) function was repeated as seen in equation 7.

$$\left. \begin{array}{l} D2 = imimposemin(D, mask); \\ Ld2 = watershed(D2); \\ bw3 = bw; \\ bw3(Ld2 == 0) = 0; \\ imshow(bw3) \end{array} \right\} \quad (7)$$

The result of the gradual controlled process of segmenting fibers from the original binary image is a new binary image that contains separated fibers [9].

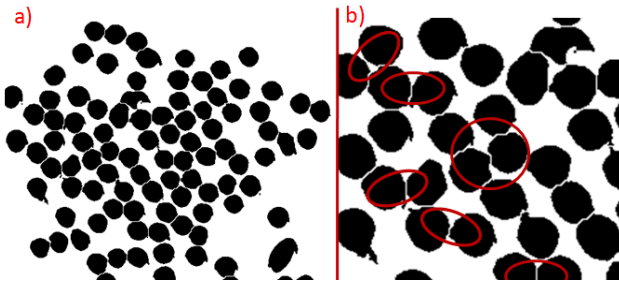


Fig. 10 a) Resulting separated fibers, b) Detailed view of separated fibers.

The division of yarn fibers into individual fibers of the examined yarn. Through this process, individual yarn fibers that were intersecting can be separated gradually or automatically. From these new images, it is now possible to generate central points. In Matlab, a script was written that allows for the marking of central points and also enables writing to the *.xls format. The result was the creation of centers of individual fibers for the first 100 CT images. See Fig 11 (a, b) [10].

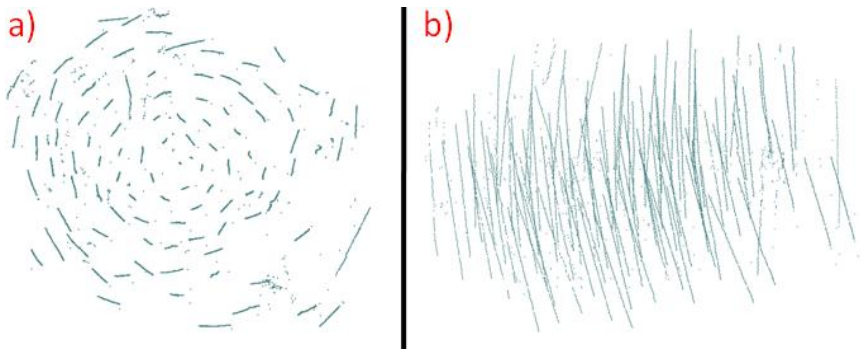


Fig. 11 a) Top view of the created point cloud, b) Isometric view of the point cloud

4 Conclusions

This work presented a comprehensive solution for the analysis of fiber images using the powerful software environment MATLAB. The main intent of the work was focused on the use of advanced algorithms for the detection, segmentation, and quantification of fibers in microscopic images of fibrous materials. The introductory part of the work concentrated on image processing, including filtering techniques, adaptive thresholding, and contrast enhancement. Subsequently, algorithms for edge detection and fiber segmentation were applied, using modern methods such as the Canny detector and Watershed transformation. The next step was to locate the central points of the already segmented fibers. These central points were then exported to an Excel file for subsequent processing in CAD software. For the quantification of fibers, geometric and textural properties such as length, diameter, and orientation of the fibers were extracted. Research in the field of fiber image analysis is a continuously developing discipline that brings new insights and application possibilities. In the future, further refinement of algorithms and methods of importing center points is expected, which will lead to even more accurate and efficient results.

References

1. N. Naouar, D. Vasiukov, C.H. Park, S.V. Lomov, P. Boisse, Meso-FE modelling of textile composites and X-ray tomography, *J. Mater. Sci.*, 55 (36) (2020), pp. 16969-16989

2. J. Angulo, S. Velasco-Forero, J. Chanussot, Multiscale stochastic watershed for unsupervised hyperspectral image segmentation IEEE International Geoscience and Remote Sensing Symposium, 3 (2009), pp. III-93
3. M. Ziabari, V. Mottaghitlab, S. T. McGovern, A. K. Haghi. Measuring Electrospun Nanofibre Diameter: a Novel Approach. *Nanoscale Res Lett*, 2, 597-600, 2007.
4. Z. Chen, R. W. Wang, X. M. Zhang, B. P. Yin. Study on Measuring Microfiber Diameter in Melt-blown Web-based on Image Analysis. *Procedia Engineering*, 15, s. 3516-3520 2011.
5. P. Arbelaez, M. Maire, C. Fowlkes, J. Malik, Contour detection and hierarchical image segmentation, *IEEE PAMI*, 33 (5) (2011), pp. 898-916
6. P. Arbeláez, J. Pont-Tuset, J.T. Barron, F. Marques, J. Malik, Multiscale combinatorial grouping, *Proceedings IEEE CVPR* (2014), pp. 328-335
7. J. Adamowski, H.F. Chan, S.O. Prasher, B. Ozga-Zielinski, A. Sliusarieva, Comparison of multiple linear and nonlinear regression, autoregressive integrated moving average, artificial neural network, and wavelet artificial neural network methods for urban water demand forecasting in Montreal Canada, *Water Resour. Res.*, 48 (2012), p. W01528
8. M.G. Schaap, W. Bouten, Modeling water retention curves of sandy soils using neural networks, *Water Resour. Res.*, 32 (1996), pp. 3033-3040
9. J. Annala, O.W. Meier, A.J. Meier, S. Grubbs GIS and artificial neural network-based water quality model for a stream network in the upper green river basin, Kentucky USA, *J. Environ. Eng.*, 04014082 (2014), pp. 1-16
10. F. Robitaille, A.C. Long, I.A. Jones, C.D. Rudd, Automatically generated geometric descriptions of textile and composite unit cells, *Compos. Appl. Sci. Manuf.*, 34 (4) (2003), pp. 303-312

Open Access This chapter is licensed under the terms of the Creative Commons Attribution-NonCommercial 4.0 International License (<http://creativecommons.org/licenses/by-nc/4.0/>), which permits any noncommercial use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.

