



OPEN

## Development and testing of a novel image analysis algorithm for descriptive evaluation of shape change of a shrinkable soft material

Pinpinat Stienkijumpai<sup>1</sup>, Maturada Jinorose<sup>1✉</sup> & Sakamon Devahastin<sup>2,3</sup>

Soft material can undergo non-uniform deformation or change of shape upon processing. Identifying shape and its change is nevertheless not straightforward. In this study, novel image-based algorithm that can be used to identify shapes of input images and at the same time classify non-uniform deformation into various patterns, i.e., swelling/shrinkage, horizontal and vertical elongations/contractions as well as convexity and concavity, is proposed. The algorithm was first tested with computer-generated images and later applied to agar cubes, which were used as model shrinkable soft material, undergoing drying at different temperatures. Shape parameters and shape-parameter based algorithm as well as convolutional neural networks (CNNs) either incorrectly identified some complicated shapes or could only identify the point where non-uniform deformation started to take place; CNNs lacked ability to describe non-uniform deformation evolution. Shape identification accuracy of the newly developed algorithm against computer-generated images was 65.88%, while those of the other tested algorithms ranged from 34.76 to 97.88%. However, when being applied to the deformation of agar cubes, the developed algorithm performed superiorly to the others. The proposed algorithm could both identify the shapes and describe their changes. The interpretation agreed well with that via visual observation.

Shape is among the most important characteristics that must be carefully designed and controlled when manufacturing a product, especially when a shrinkable soft material is used as a starting raw material<sup>1</sup>. This is because such a material consists mainly of water, which may need to be removed (or, in other words, dehydrated) during processing. Material structure, e.g., cell wall of plants or even soft structure of various gels, which holds and is indeed supported by water would then collapse, resulting in shrinkage and deformation<sup>2–6</sup>. Since moisture gradients generally exist during dehydration due to different rates of moisture removal at different locations, deformation is in most cases non-uniform<sup>2,7</sup>. This leads in turn to shape change of a resulting product, which may be either desirable or undesirable, depending on a specific manufacturing situation and purpose.

Non-uniform deformation is of interest as such a deformation would lead to changes in both size and shape of a product undergoing processing. These changes in turn affect heat and mass transfer behavior as well as many other properties of a product<sup>8–10</sup>. A means to accurately characterize and describe this type of deformation is therefore highly desirable. Deformation is nevertheless typically determined and reported in terms of volumetric shrinkage, which is a ratio of volume change of a specimen to its initial volume<sup>11,12</sup>. However, volumetric shrinkage cannot clearly be used to quantify the change of shape (i.e., non-uniform deformation). It has indeed been reported that the same specimen undergone two different processes and/or conditions could have the same volumetric shrinkage but completely different shapes<sup>11,13</sup>.

Image analysis is an alternative method that can be used for material and product characterization; the method can well be used for the determination of both size and shape of a product<sup>14,15</sup>. Nevertheless, although

<sup>1</sup>Department of Food Engineering, School of Engineering, King Mongkut's Institute of Technology Ladkrabang, 1 Soi Chalong Krung 1, Ladkrabang, Bangkok 10520, Thailand. <sup>2</sup>Advanced Food Processing Research Laboratory, Department of Food Engineering, Faculty of Engineering, King Mongkut's University of Technology Thonburi, 126 Pracha u-tid Road, Tungkru, Bangkok 10140, Thailand. <sup>3</sup>The Academy of Science, The Royal Society of Thailand, Dusit, Bangkok 10300, Thailand. ✉email: maturada.ji@kmitl.ac.th

Deformation pattern	Number of images		% Accurate identification						
			Without alignment	With alignment			Igathinathane et al. <sup>12</sup>	CNN	
				EEM	Extrema	Manual		ResNet-50	Xception
0, 0, 0, 0, 0	120		35.00	47.50	100.00	86.67	40.00	100.00	100.00
+, +, +, 0, 0	240		32.50	46.67	85.00	83.33	40.00	100.00	100.00
+, +, 0, 0, 1	240		32.08	40.00	70.00	98.33	40.00	100.00	100.00
+, 0, +, 0, 1	240		32.50	59.58	82.92	100.00	40.00	100.00	100.00
0, 0, 0, -, 1	240		30.42	44.17	92.92	74.17	41.67	100.00	100.00
+, +, +, -, 1	480		35.21	45.00	85.83	82.08	41.04	100.00	100.00
+, +, 0, -, 1	480		33.96	44.17	68.13	87.50	35.83	100.00	100.00
+, 0, +, -, 1	480		32.92	60.00	80.83	86.88	26.46	99.79	100.00
0, 0, 0, +, 1	240		27.92	40.42	50.00	67.50	20.00	41.67	84.58
+, +, +, +, 1	480		26.04	39.58	48.54	60.63	20.00	100.00	100.00
+, +, 0, +, 1	480		25.83	28.96	42.71	61.42	40.00	52.08	92.50
+, 0, +, +, 1	480		30.63	56.88	35.00	62.71	40.00	50.21	96.67
	4200	Avg	30.98	45.93	65.88	77.10	34.76	85.48	97.88

**Table 1.** Accuracies of various shape identification algorithms. EEM = equivalent ellipsis method.

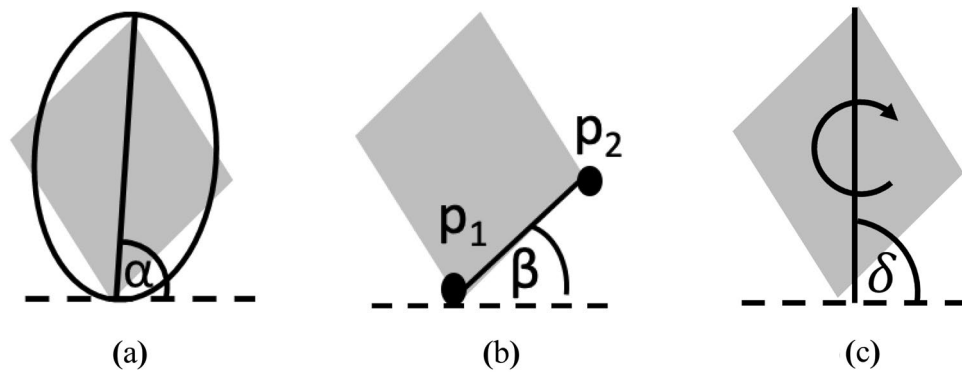
there exist several algorithms that can be used to identify shape, e.g., template matching method<sup>14</sup>, or such statistical methods as k-cluster and discriminant analyses<sup>14,16</sup> as well as decision tree<sup>14,17–19</sup>, such available algorithms can only classify images into a limited number of predetermined shapes, making these algorithms less robust and flexible. The limitation exists because in order to evaluate shape using image analysis, appropriate shape parameters must first be devised and tested. Although various shape parameters have been proposed<sup>9,14,17</sup> most reported works were performed using shape parameters that can only describe or identify some specific shapes. In other words, each parameter has its own set of shapes that can be identified; therefore shape must be indicated a priori<sup>20</sup>, which is clearly an inappropriate approach. Most available algorithms cannot also be used to describe non-uniform deformation; only a period over which non-uniform deformation takes place could be identified. In addition, in a real-world situation, a product entering an image analysis system may be arbitrarily oriented<sup>17,18,21</sup>; this may lead to an inaccurate decision during the shape matching process. Therefore, an input image needs to be properly aligned prior to being analyzed for more accurate results<sup>22</sup>. Most existing algorithms lack this important image processing step, however.

More robust neural network-based algorithms such as convolutional neural networks (CNNs) have recently been used to identify shapes of input images. CNNs have successfully been used, for example, to identify shapes of crystal structures<sup>23,24</sup>. Performance of CNNs nevertheless depends on their architectures; various architectures, including GoogLeNet, ResNet-50, Xception, VggNet-19, AlexNet and EfficientNet-16 have been studied<sup>25,26</sup>. More importantly, for CNNs to succeed in real-world situations, a large number of images are required to train the networks. Such a training (and subsequent validation) step may require large computational resources and time, hence preventing the use of CNNs in a situation where limited developmental time and resources are available. Acquiring a sufficient number of real-world images may also sometimes not be feasible; this may in turn adversely affect the identification accuracy<sup>27</sup>. More importantly, most available algorithms cannot be used to describe the evolution of non-uniform deformation; only the shape (and not its evolution) can be described and quantified.

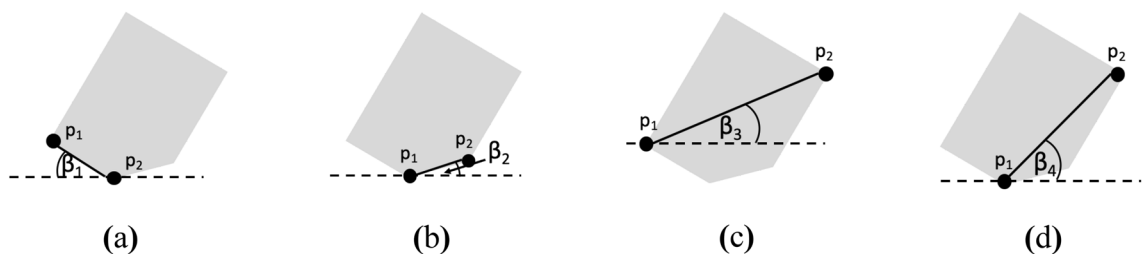
The present study aimed at developing a novel image-based algorithm that can be used to determine the shapes of input images and at the same time capture changes in the shapes (non-uniform deformation) of such images. The effects of alignment and alignment methods on the resulting shape analysis results were investigated. Newly proposed scheme that can be used to classify non-uniform deformation into various patterns, i.e., swelling/shrinkage, horizontal and vertical elongations/contractions as well as convexity and concavity, is also proposed. The developed algorithm was first tested with computer-generated images and later applied to cubes of agar gel of different compositions, which were used as a model highly shrinkable soft material, undergoing drying at different temperatures. Analysis results were compared with those obtained using shape parameters and shape-parameter based algorithm as well as CNNs.

## Results and discussion

**Test of shape identification algorithms.** First, the accuracies of the newly proposed shape identification procedures with and without image alignment were compared; the results are shown in Table 1. The results of the use of the algorithm by Igathinathane et al.<sup>17</sup> and CNNs are also listed. Note that the shape identification accuracy was calculated by comparing the known shapes of the computer-generated images with those identified by each tested algorithm. A count was made on both correct and incorrect shape identifications; percent accuracy was then calculated. In the case of the newly developed algorithm, identifying the shape of input images without any alignment yielded the lowest accuracy, as expected. Unaligned images had higher chances of being overlapped with other reference geometries than with the most suitable ones. Nevertheless, although the alignment step could help improve the accuracy of shape identification, such a step resulted in a longer processing time. Zigzagging artefact, which was the results of the alignment step<sup>28</sup>, might also affect shape identification.



**Figure 1.** Image alignment methods. (a) Equivalent ellipsis method, (b) extrema method and (c) manual alignment method.



**Figure 2.** Possible  $\beta$  angles to refer to during alignment using extrema method.

Different alignment methods (see Fig. 1) yielded different levels of accuracy as shown in Table 1. Equivalent ellipsis method resulted in the lowest accuracy. When images were not deformed (0, 0, 0, 0, 0), major axis of those images could have already been aligned, either on the horizontal or vertical axis, causing the method not to rotate the images until reaching  $\alpha = 0^\circ$  or even rotating the images upside-down; this is because  $\alpha$  is always the acute angle between the major axis and horizontal axis. When applying the equivalent ellipsis method to images with horizontal elongation (+, +, 0, 0, 1), the method yielded the lowest accuracy. For example, when a triangle or pentagon was horizontally elongated, the method identified it instead as a rectangle or hexagon. Such an error occurred because the apex angle would change from smaller than  $90^\circ$  to larger than  $90^\circ$ . The method performed better when images were vertically elongated (+, 0, +, 0, 1) because the major axis of those images would align on the longest axis of the images. Nevertheless, this alignment method required the shortest processing time (around 0.25 s per image using the currently employed hardware) because it refers to only one angle (a) in order to perform the alignment process.

Extrema method was noted to give more accurate identification results than the equivalent ellipsis method. The accuracy was higher than 50% if input images resembled their reference geometric shapes. This is because the extrema points of the images would be the same as those of the reference geometric images. On the other hand, when images were convexed (0, 0, 0, +, 1), the extrema points sometimes changed to the convex points. Convexity also caused images to look more like a circle or an ellipse. This alignment method required longer processing time than the equivalent ellipsis method (around threefold longer) as the former needs to refer to up to four angles to perform the alignment process (see Fig. 2).

Manual alignment method gave the highest accuracy (higher than 60% in all cases). The major advantage of this method arose when images experienced horizontal (+, +, 0, 0, 1) or vertical elongation (+, 0, +, 0, 1); such images would suffer less inaccuracy than the ones with no deformation (0, 0, 0, 0, 0). This is because the manual alignment method rotates an image from  $0^\circ$  to  $360^\circ$  at  $1^\circ$  interval, causing some zigzagging artefact on the image edge. However, when images were elongated, the dimension ratio of the artefact to the characteristic length of the images would decrease. Nevertheless, this method required the longest time, 51.29 s per image or around 70-fold longer than the extrema method. This is expected as it has 360 possible angles to refer to.

Although the manual alignment method resulted in the highest accuracy, the processing time was excessive. This is clearly not appropriate, especially if real-time image-based process control is to be conducted. For this reason, extrema method was chosen because it exhibited higher accuracy than the equivalent ellipsis method and yet performed 70 times faster than the manual alignment method.

It is seen in Table 1 that the developed algorithm (employing the extrema alignment method) gave 100.00% and 92.92% accuracy when input images did not suffer any deformation (0, 0, 0, 0, 0) and when images were concaved (0, 0, 0, -, 1), respectively. This is because the points at the angles of those images were distinct (except in the case of circles). This in turn caused the RMSD values to be lower. Swollen images (+, +, 0, 0) having more zigzagging edges, caused by the forced deformation process, would, for example, result in pentagons or hexagons

looking more like circles. Horizontal elongation exhibited stronger adverse effect than vertical elongation because when images with apex angles of less than  $90^\circ$  (such as triangles and pentagons) were horizontally elongated, the angles would become larger than  $90^\circ$ . This in turn affected RMSD because this value was calculated from  $d$  values at the same angle  $\theta$ . The algorithm did not well identify images when they were convexed as the image would become more like circles.

When all 4200 computer-generated images were analyzed using the developed algorithm with the extrema method, the overall capability to accurately identify the shapes of input images was noted to be 65.88%. In contrast, when the algorithm by Igathinathane et al.<sup>17</sup> and CNN with ResNet-50 architecture (CNN ResNet-50) and CNN with Xception architecture (CNN Xception) were used for the same shape identification purpose, the overall accuracies were noted to be 34.76%, 85.48 and 97.88% respectively. The reason for the much lower overall accuracy of the algorithm by Igathinathane et al.<sup>17</sup> is that the algorithm could not identify pentagons and hexagons as these shapes are not included in the algorithm; computer-generated pentagons and hexagons were therefore incorrectly identified as either ellipses or rectangles. This clearly indicates the disadvantage of this (or similar algorithms) where new reference geometric shapes always need to be added and analyzed when dealing with input images of peculiar shapes. In the case of CNNs, the algorithms were not good at identifying both horizontally and vertically convexed images. This may probably be due to the inherently ambiguous sorting classes and blurring appearance<sup>29</sup>. CNN Xception performed better than CNN ResNet-50, as expected<sup>30</sup>. It is important to note, however, that both the newly developed algorithm and that of Igathinathane et al.<sup>17</sup> require no training, while CNN ResNet-50 and CNN Xception require significant training time. Using the currently employed hardware, as much as 41,942 and 138,269 s were required, respectively, for the two CNNs.

**Comparative performance of developed algorithm and existing shape parameters.** *Effect of size change.* The effect of the change in image size on the Extent was first evaluated. Size change did not significantly affect the Extent. This is because when the size of an image increased, the size of the rectangular bounding box also increased at the same rate. Each geometric shape was noted to exhibit its own Extent value; the values for a triangle, rectangle, pentagon, hexagon and circle are  $0.525 \pm 0.003$ ,  $1.000 \pm 0.000$ ,  $0.710 \pm 0.002$ ,  $0.768 \pm 0.002$  and  $0.805 \pm 0.004$ , respectively. Extent nevertheless has a significant disadvantage that it can only be used to identify reference geometric shapes. This shape parameter cannot be used to describe non-uniform deformation; only a point where non-uniform deformation starts to take place could be identified<sup>15</sup>.

In the case of fractal dimension, change of this parameter could be divided into three periods based on the ratio of area of interest (AOI) to the total area of an image. When the aforementioned ratio was lower than 5%, fractal dimension was close to unity. This unexpected result was noted because the method that was used to calculate the fractal dimension here was the box counting method. When the size decreased, AOI resembled a point. This implies that fractal dimension calculated by the box counting method could not be used when the ratio of AOI to the total area of an image is lower than 5%. On the other hand, when such a ratio was higher than 5% but lower than 20%, fractal dimensions of a triangle, rectangle, pentagon, hexagon and circle were noted to be  $1.826 \pm 0.061$ ,  $1.863 \pm 0.036$ ,  $1.852 \pm 0.052$ ,  $1.869 \pm 0.051$ ,  $1.883 \pm 0.051$ , respectively. Finally, when such a ratio was higher than 20%, all fractal dimensions approached 2.

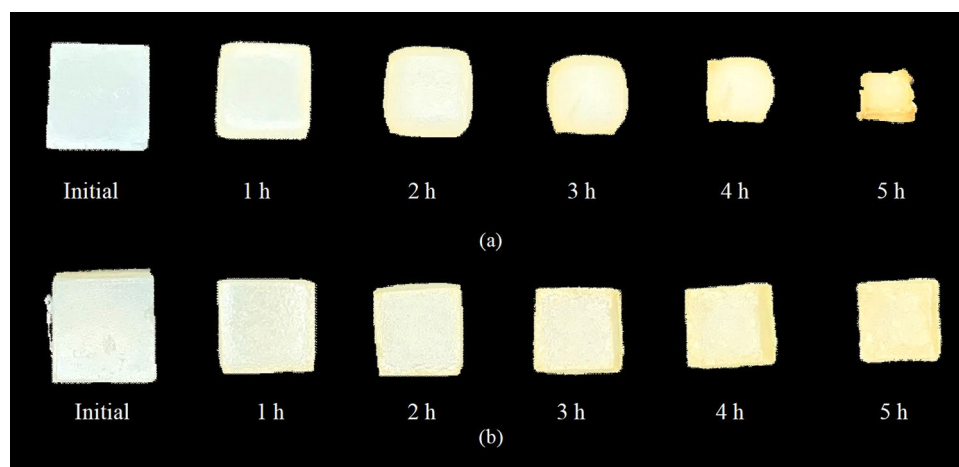
In the case of the presently developed algorithm, the change in size affected  $i$ ,  $j$  and  $k$ . This is expected as  $i$  is related to image area, which is in turn related to size;  $j$  and  $k$  are also linearly related to horizontal and vertical dimension changes. Parameter  $l$ , which is used to indicate convexity and concavity was not affected by the change in size, as expected.

*Effects of horizontal and vertical length changes.* Both horizontal elongation and contraction did not affect the Extent because these changes equally affected both AOI and its rectangular bounding box. In the case of the newly developed algorithm, horizontal elongation and contraction affected  $i$  and  $j$  but did not affect  $k$  and  $l$ . When an image was horizontally elongated or contracted, its size also increased or decreased, respectively. However, since the change occurred only in the horizontal direction,  $k$  was not affected. Parameter  $l$  was also not affected as no convexity took place. The above-mentioned reasons can also be used to explain the effect of vertical length change.

*Effects of convexity and concavity.* Extent of a triangle was most significantly affected and so can be used to quantify both the convexity and concavity. In contrast, the Extent of a rectangle exhibited similar patterns of changes, whether convexity or concavity took place; Extent cannot therefore be used to classify convexity and concavity of this geometric shape. Extent values of a concaved pentagon, hexagon and circle exhibited similar patterns; however, the values belonging to these shapes that were made to convex remained almost unchanged. Therefore, the Extent can only be used to identify concavity of these shapes.

Since each geometric shape exhibited its own Extent value, convexity and concavity of a triangle could, for example, be identified by looking at the deviation of its Extent value from the reference value of 0.525; higher Extent value than the reference value indicated convexity and vice versa. Similar identifications can also be made for a pentagon, hexagon and circle. Since convexity and concavity caused the image perimeter to be rougher, fractal dimension increased. However, this parameter cannot be used to describe such a change of shape, only being able to identify if any change exists.

In the case of the presently developed algorithm, concavity and convexity only affected parameter  $l$ . As a result, this parameter can be used to identify and quantify convexity and concavity by looking at the deviation of its value from zero in much the same way as in the case of Extent. However, this parameter can be applied to all tested shapes, including a rectangle.



**Figure 3.** Deformation of agar cubes with (a) 0% sugar and (b) 20% sugar during drying at 80 °C.

Moisture ratio	1.00 ± 0.00	0.78 ± 0.02	0.56 ± 0.03	0.36 ± 0.03	0.20 ± 0.02	0.06 ± 0.01
0% sucrose						
	Shape = Rectangle i = 100 j = 100 k = 100 l = 100 m = "Uniform" Shape <sub>ig</sub> = Rectangle Shape <sub>CNN</sub> = Rectangle Extent = 0.9473 D <sub>F</sub> = 2.1399	Shape = Rectangle i = 84.35 j = 96.84 k = 96.93 l = 101.11 m = "Non-uniform" Shape <sub>ig</sub> = Inclined rectangle Shape <sub>CNN</sub> = Rectangle Extent = 0.9396 D <sub>F</sub> = 2.1171	Shape = Rectangle i = 68.69 j = 84.98 k = 83.91 l = 102.76 m = "Non-uniform" Shape <sub>ig</sub> = Inclined rectangle Shape <sub>CNN</sub> = Rectangle Extent = 0.9167 D <sub>F</sub> = 2.0961	Shape = Hexagon i = 52.75 j = 77.47 k = 75.48 l = 103.32 m = "Non-uniform" Shape <sub>ig</sub> = Inclined rectangle Shape <sub>CNN</sub> = Rectangle Extent = 0.8771 D <sub>F</sub> = 2.0747	Shape = Pentagon i = 36.84 j = 65.22 k = 60.92 l = 102.15 m = "Non-uniform" Shape <sub>ig</sub> = Inclined rectangle Shape <sub>CNN</sub> = Rectangle Extent = 0.8845 D <sub>F</sub> = 2.0545	Shape = Pentagon i = 24.07 j = 58.01 k = 50.19 l = 99.33 m = "Non-uniform" Shape <sub>ig</sub> = Inclined rectangle Shape <sub>CNN</sub> = Rectangle Extent = 0.7944 D <sub>F</sub> = 2.0309
20% sucrose						
	Shape = Rectangle i = 100 j = 100 k = 100 l = 100 m = "Uniform" Shape <sub>ig</sub> = Inclined rectangle Shape <sub>CNN</sub> = Rectangle Extent = 0.8637 D <sub>F</sub> = 2.2786	Shape = Rectangle i = 79.27 j = 86.62 k = 84.81 l = 100.04 m = "Uniform" Shape <sub>ig</sub> = Inclined rectangle Shape <sub>CNN</sub> = Rectangle Extent = 0.8613 D <sub>F</sub> = 2.2423	Shape = Rectangle i = 69.24 j = 81.41 k = 81.27 l = 100.70 m = "Uniform" Shape <sub>ig</sub> = Inclined rectangle Shape <sub>CNN</sub> = Rectangle Extent = 0.8627 D <sub>F</sub> = 2.2508	Shape = Rectangle i = 62.40 j = 80.67 k = 82.33 l = 99.98 m = "Uniform" Shape <sub>ig</sub> = Inclined rectangle Shape <sub>CNN</sub> = Rectangle Extent = 0.8712 D <sub>F</sub> = 2.2615		

**Table 2.** Top-view images of agar cubes with 0 and 20% sucrose during drying at 80 °C. Shape<sub>CNN</sub> was that obtained via CNN Xception.

Note that parameter *m* is used to identify if deformation is uniform or non-uniform. Such a decision is made based on the values of *a* and *l*. Uniform deformation is said to take place only if both the value of *a* is higher than 95% and the value of *l* is higher than 99%. The chosen threshold of 95% is based on the typical confidence level used in most engineering statistical analysis. The threshold of 99% was, on the other hand, chosen by adjusting the value until the computed identification matched that visually observed.

**Test of developed algorithm on agar cubes.** Performance of the developed algorithm was evaluated by applying it to agar cubes containing either 0 or 20% sucrose undergoing hot air drying at 80 °C (see Fig. 3). Top-view images of agar cubes containing 0 and 20% sucrose are shown in Table 2. The algorithm indicated that the top-view shape of agar cubes with 0% sucrose changed from square to rectangle to hexagon when the moisture ratios were about 0.78 and 0.36, while no shape change was detected in the case of agar cubes with 20% sucrose. Agar with 0% sucrose had almost no solids to support its structure after water had been evaporated. On the other hand, agar with 20% sucrose possessed more solids to support its structure and hence deformed more uniformly. In

other words, the cubes with 20% sucrose experienced only the change in size; no convexity and concavity were observed.

Top-view shape changes of agar cubes with 0% sucrose could be divided into three periods. First, the shape of the cubes was slightly convexed. This is because their edges were hardened while the other portions, which were still moist and elastic, were forced to convexed out (see the results at the moisture ratios of 0.78 and 0.56). *i* decreased from 100 to 84.35 and 68.69% when the area decreased by 15.65 and 31.31%, while *j* decreased from 100 to 96.84 and 84.98%, indicating the horizontal contractions of 3.16 and 15.02%. *k* also decreased from 100 to 96.93 and 81.91%, indicating the vertical contractions of 3.07 and 18.09%. The decreasing rate of *k* was higher than that of *j*, implying that the vertical contraction was more extensive than the horizontal one. This is because the direction of drying air flow in the present experimental study was perpendicular to the top horizontal edges of the cubes. In the case of *l*, its value increased from 100 to 101.11 and 102.76%, indicating, as mentioned earlier, that convexity had occurred. The top-view shape of the cubes then appeared as rectangle rather than as square. Parameter *m* also correctly indicated the non-uniform deformation.

During the second period, when the moisture ratios were between 0.56 and 0.36, the shape of the cubes changed into hexagon. *i* decreased from 68.69 to 52.75%, indicating that shrinkage still occurred. The value of *j* decreased from 84.98 to 77.47%, while that of *k* decreased from 83.91 to 75.48%. Trends of changes of *j* and *k* were similar to those during the first period in that there was more extensive vertical contraction than the horizontal one. The value of *l* changed from 102.76 to 103.32%, indicating that there was also convexity. It was indeed this convexity that led to further shape change. Parameter *m* again indicated non-uniform deformation.

Finally, when the moisture ratios were lower than 0.36, structural collapse was noted, resulting in the change of shape of agar cubes from hexagon to pentagon. This is because agar cubes exhibited concavity in the middle softest region prior to finally being hardened (see the results at the moisture ratios of 0.20 and 0.06). The value of *i* decreased to 24.07%, implying that agar cubes extensively shrunk; *j* and *k* also decreased to 58.01 and 50.19%, respectively. The value of *l* decreased to 99.33%, implying that the cubes were concaved. Parameter *m* again indicated that the deformation was non-uniform in nature.

The results obtained by the developed algorithm were compared with those obtained using the algorithm developed by Igathinathane et al.<sup>17</sup> and CNN Xception, in terms of Shape<sub>ig</sub> and Shape<sub>CNN</sub>, respectively; CNN Xception was selected due to its higher overall identification accuracy. The results obtained using Extent and fractal dimension were also compared. The results are as shown in Table 2. The developed algorithm could both identify and describe the changes of shape. The interpretation indeed agreed with that via visual observation. On the contrary, the algorithm of Igathinathane et al.<sup>17</sup> could only identify the shape as “inclined rectangle” even though the cubes changed into other shapes (e.g., pentagon, hexagon) rather than rectangle. Interestingly, although CNN Xception could identify the shapes of computer-generated images very accurately, it could only identify the shape of agar cubes as “rectangle.” This might be due to the insufficient number of images for the network training. Although as many as 45,000 images were used for training the network, it still did not succeed in this real-world application. This illustrates a limitation of CNNs when being applied even to identify simple shapes of practical materials and/or objects. Acquiring a sufficient number of real-world images for training may not always be feasible, however.

Extent and fractal dimension could only identify the points where non-uniform deformation started to take place but lacked the ability to describe such a deformation. This agrees with the observation reported by Jinorose et al.<sup>15</sup>.

## Conclusion

A novel image analysis algorithm that can be used to identify shape and describe non-uniform deformation of an input image is proposed and was tested with both computer-generated images and cubes of agar gel, which was used as the test highly shrinkable material, undergoing drying. Identifying the shape of an input image without any alignment yielded the lowest accuracy. Among the tested alignment methods, the extrema method performed adequately when both the efficiency and required computational resource were taken into consideration. The algorithm was well capable of distinguishing the different deformation patterns of the computer-generated images and, in particular, agar cubes with different solids contents. In the latter case, the developed algorithm was superior to the use of simple shape parameters, shape-parameter based algorithm and even convolutional neural networks. It is important to note that the developed algorithm can identify any shape and its change without being limited to only predetermined shapes as in the case of other existing algorithms. The developed algorithm is robust and could be further developed for real-time process control of a drying process where size and shape of a material, in particular soft material, are of concern.

## Algorithm development and testing

**Algorithm development.** *Computer specification.* Personal computer with Intel® Core™ i7-10657G7 at 1.30 GHz was used in all cases. Computer is installed with NVIDIA GeForce GTX 1650 with Max-Q Design as Graphics Processing Unit with 32 GB of RAM and 512 GB of SSD. The operating system is Windows 10 Home.

*Image pre-processing steps.* Computer-generated images were first segmented using the algorithm described by Jinorose et al.<sup>15</sup>. First, artefact was eliminated from each image via the use of MATLAB® (version R2020a, MathWork Inc., MA). Each image was then cropped to 512 × 512 pixel. Image segmentation was performed by converting an RGB image into a binary image using Otsu’s thresholding method<sup>15</sup>. Edge detection and holes filling were subsequently performed to extract an area of interest.

As in reality a material entering an image analysis system may initially be arbitrarily oriented, a material image must first be aligned into an axis prior to being analyzed. Three alignment methods were tested in this



**Figure 4.** Reference geometric shapes used in this study.

study viz. equivalent ellipsis, extrema and manual alignment methods (see Fig. 1). In the case of the equivalent ellipsis method, image would be rotated until  $\alpha = 0^\circ$ . Note that  $\alpha$  is an angle between the major axis of an ellipse that has the same second moment of area of interest as that of an input image and the horizontal axis<sup>31</sup> (Fig. 1a). In the case of the extrema method, image would be rotated until  $\beta = 0^\circ$ . In this case,  $\beta$  is an angle between the line connecting points  $p_1$  and  $p_2$ , which are the two extrema points, and the horizontal axis<sup>28</sup> (Fig. 1b). In the case of the manual alignment method, image would be rotated from  $0^\circ$  to  $360^\circ$  at every  $1^\circ$  interval. The final angle of rotation ( $\delta$ ) was selected as the one that gave the lowest root mean square difference (RMSD) of the distance between the centroid and the edge of the image at angle  $\theta$  ( $d_{\text{image},\theta}$ ) and that of the reference standard geometric shape ( $d_{\text{ref},\theta}$ ) at the same angle  $\theta$  (Eq. 1).

$$\text{RMSD} = \sqrt{\frac{\sum_{\theta=0}^{360} (d_{\text{ref},\theta} - d_{\text{image},\theta})^2}{360}} \quad (1)$$

The final angle of rotation was then used to rotate the image as per the following rotation matrix (Eq. 2) using the 'affine2d' and 'imwarp' functions in MATLAB R2020a<sup>31</sup>:

$$\text{Rotation matrix} = \begin{bmatrix} \cos & -\sin & 0 \\ \sin & \cos & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (2)$$

*Matching input image with reference geometric shapes.* Five reference geometric shapes were used in this study, i.e., triangle, rectangle, pentagon, hexagon and circle (Fig. 4). Each image itself was with the dimensions of  $97 \times 97$  pixel, while the black background where the image sat on was with the dimensions of  $512 \times 512$  pixel. An input image was matched to one of these geometric shapes to identify the closest shape of the image.

To identify the shape of an input image, its bounding box would be expanded until the length and width of that bounding box equal to those of the box belonging to all the reference geometric shapes. RMSD was then calculated and the most appropriate shape was selected as the one giving the lowest RMSD value.

*Description of non-uniform deformation of input image.* Before being able to describe non-uniform deformation of an input image, reference geometric shapes were first made to deform in various ways. The input image was first compared with the one with no deformation. The deformation patterns include swelling and shrinkage, horizontal and vertical elongations/contractions as well as convexity and concavity.













Swelling and shrinkage (or negative swelling) were achieved as per the swelling matrix (Eq. 3)<sup>14</sup>. Elongation, both in horizontal and vertical directions, was achieved as per the elongation matrices (Eqs. 4 and 5)<sup>14</sup>, while convexity and concavity functions are defined as shown in Eqs. (6) and (7)<sup>32</sup>. Built-in 'affine2d' and 'geometricTransform2d' functions were used along with the above-mentioned equations to achieve the deformation; 'imwarp' function was used to implement all the deformation operations.

$$\text{Swelling matrix} = \begin{bmatrix} S & 0 & 0 \\ 0 & S & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (3)$$

$$\text{Horizontal elongation matrix} = \begin{bmatrix} S & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (4)$$

$$\text{Vertical elongation matrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & S & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (5)$$

$$d_{\text{convex}} = d_{\text{image},\theta} + d_{\text{image},\theta}^3 \left( \frac{S}{\max(d_{\text{image},\theta})^2} \right) \quad (6)$$

Deformation pattern	+	0	-
Swelling and shrinkage			
i, j, k, l, m system	+, +, +, 0, 0	0, 0, 0, 0, 0	-, -, -, 0, 0
Horizontal elongation and contraction			
i, j, k, l, m system	+, +, 0, 0, 1	0, 0, 0, 0, 0	-, -, 0, 0, 1
Vertical elongation and contraction			
i, j, k, l, m system	+, 0, +, 0, 1	0, 0, 0, 0, 0	-, 0, -, 0, 1
Convexity and concavity			
i, j, k, l, m system	0, 0, 0, +, 1	0, 0, 0, 0, 0	0, 0, 0, -, 1

**Table 3.** Deformation patterns of reference geometric shapes.

$$d_{\text{concave}} = d_{\text{image},\theta} - d_{\text{image},\theta}^3 \left( \frac{S}{\max(d_{\text{image},\theta})^2} \right) \tag{7}$$

where  $S$  is the scaling factor.

All the deformation patterns are summarized in Table 3. Parameters  $i, j, k, l$  and  $m$  are used to describe the various deformation patterns. Parameter  $i$  (Eq. 8) indicates swelling and shrinkage; ‘+’ indicates swelling, while ‘-’ indicates shrinkage. Parameter  $j$  (Eq. 9) indicates horizontal elongation (+) and contraction (-). Parameter  $k$  (Eq. 10) indicates vertical elongation (+) and contraction (-). Parameter  $l$  (Eq. 11) indicates convexity (+) and concavity (-). Finally, parameter  $m$  is used to indicate whether the deformation is uniform or not. This latter parameter was calculated from Eq. (13) and the aforementioned parameter  $l$ ; when  $\alpha$  is less than 0.05 and  $l$  is less than 0.01,  $m$  would indicate uniform deformation. The choices of these decision criteria is as explained in the “Results and discussion” section.

$$i = \frac{A - A_{\text{ini}}}{A_{\text{ini}}} \tag{8}$$

$$j = \frac{L - L_{\text{ini}}}{L_{\text{ini}}} \tag{9}$$

$$k = \frac{W - W_{\text{ini}}}{W_{\text{ini}}} \tag{10}$$

$$l = \frac{\text{Extent}_{\text{Ellipse}} - \text{Extent}_{\text{Ellipse,ini}}}{\text{Extent}_{\text{Ellipse,ini}}} \tag{11}$$

$$\text{Extent}_{\text{Ellipse}} = \frac{A}{\frac{\pi}{4}(\text{major})(\text{minor})} \tag{12}$$

$$\alpha = \frac{j - k}{\min(j, k)} \times 100 \tag{13}$$

In the above equations,  $A$  is the projected area of the area of interest (AOI),  $L$  is the length of the bounding box of AOI,  $W$  is the width of the bounding box of AOI. ‘major’ and ‘minor’ are the lengths (in pixel) of the major axis and minor axis, respectively, of the ellipse that has the same normalized second central moment as that of AOI.

*Comparison between developed and existing algorithms.* Performance of the developed algorithm was compared with that of algorithm developed by Igathinathane et al.<sup>17</sup>. These investigators developed the algorithm that could identify an image into either triangle, rectangle or circle. Three shape parameters, i.e., reciprocal aspect ratio (RAR), rectangularity as well as Feret major ratio (FMR), which were calculated as per Eqs. (14–16),



were used for such an identification purpose. Each of these parameters was assigned a specific range over which each of the three shapes ( $\text{Shape}_{\text{ig}}$ ) would be identified. Note that if the shape of an input image was far from the above three standard shapes, the algorithm would fail to capture such a shape; all peculiar shapes would be identified as an ‘inclined rectangle.’

$$\text{RAR} = \frac{\text{minor}}{\text{major}} \quad (14)$$

$$\text{Rectangularity} = \frac{\frac{\pi}{4}(\text{major})(\text{minor})}{A_{\text{BB}}} \quad (15)$$

$$\text{FMR} = \frac{D_f}{\text{major}} \quad (16)$$

where  $A_{\text{BB}}$  is the area of a rectangular bounding box, while  $D_f$  is the maximum Feret diameter.

For non-uniform deformation description, Extent (Eq. 17)<sup>15,31</sup> and fractal dimension (Eq. 18)<sup>33</sup> were also used. The description results based on these parameters were compared with those given by the newly developed algorithm.

$$\text{Extent} = \frac{A}{A_{\text{BB}}} \quad (17)$$

$$\text{Fractal dimension}(D_F) = \frac{\log(N_r)}{\log(\frac{1}{r})} \quad (18)$$

where  $N_r$  is the number of boxes intercepted with the size of each iteration of  $r$  value, while  $r$  is the size of the box.

Finally, convolutional neural networks (CNNs) with two different architectures, i.e., ResNet-50 and Xception, were tested via the use of the Deep Learning toolbox in MATLAB® (version R2020a, MathWork Inc., MA). Stochastic gradient descent with momentum (SGDM) with InitialLearnRate of 0.01 were used<sup>34</sup>. The validation frequency was 315, while the MaxEpochs and MinBatchSize were both set at 10.

In the cases of the presently developed algorithm and that of Igathinathane et al.<sup>17</sup> 4200 computer-generated images were created and used to test the algorithms. On the other hand, in the case of CNNs, 45,000 computer-generated images were created and used to train (31,500 images) and validate (13,500 images) the networks. The same 4200 images were also used to test the trained CNNs.

**Agar drying.** *Agar preparation.* 2% (w/w) granulated purified agar (Product no. 1016141000, Merck Millipore Corp., Darmstadt, Germany) was mixed with either 0 or 20% (w/w) sucrose solution. The mixture was stirred at room temperature ( $25 \pm 2^\circ\text{C}$ ) at 100 rpm for 1 h, then heated to  $95^\circ\text{C}$  and finally stirred at 150 rpm for 10 min. The mixture was cooled at room temperature until its temperature reached  $50^\circ\text{C}$ . The cooled mixture was poured into a silicone mold to form agar cubes, each with the dimensions of  $1.9 \times 1.9 \times 1.9$  cm. The sample was allowed to set at room temperature for 1 h before being kept at  $4 \pm 1^\circ\text{C}$  for 24 h.

*Drying experiments.* Drying experiments were conducted in a hot air dryer (Memmert GmbH+ Co. KG, UM500, Schwabach, Germany) at  $80^\circ\text{C}$ . Three agar cubes were taken out for moisture content determination and image acquisition at every 1 h.

*Moisture content determination.* A sample was weighed using a digital balance with an accuracy of  $\pm 0.0001$  g (Sartorius Lab Instruments GmbH & Co. KG, BSA224S-CW, Göttingen, Germany) and then dried in a hot air oven (Memmert GmbH+ Co. KG, UM500, Schwabach, Germany) until constant mass was obtained as per AOAC method 984.25 (2000). The moisture content (MC) of the sample was calculated using Eq. (19):

$$\text{MC}(\% \text{d.b.}) = \frac{m_i - m_{\text{bd}}}{m_{\text{bd}}} \times 100\% \quad (19)$$

where  $m_i$  is the mass of agar before drying at  $105^\circ\text{C}$ , while  $m_{\text{bd}}$  is the bone-dry mass.

*Image acquisition and segmentation.* The utilized image acquisition system was the one described by Jinorose et al.<sup>15</sup>. Only top-view images were acquired and analyzed. Image segmentation was implemented as also described by Jinorose et al.<sup>15</sup>.

## Data availability

Experimental and relevant data are available from the authors upon suitable request.

Received: 23 June 2021; Accepted: 17 August 2021

Published online: 13 September 2021

## References

1. Yang, Y. *et al.* Application of feedback control and in situ milling to improve particle size and shape in the crystallization of a slow gluing needle-like active pharmaceutical ingredient. *Int. J. Pharm.* **533**, 49–61 (2017).
2. Aguilera, J. S. & Stanley, D. W. *Microstructural Principles of Food Processing and Engineering* (Aspen Publication, 1999).
3. Mao, B., Divoux, T. & Snabre, P. Impact of saccharides on the drying kinetics of agarose gels measured by in-situ interferometry. *Sci. Rep.* **7**, 41185 (2017).
4. Khan, M. I. H., Farrell, T., Nagy, S. A. & Karim, M. A. Fundamental understanding of cellular water transport process in bio-food material during drying. *Sci. Rep.* **8**, 15191 (2018).
5. Lozano, P. F. *et al.* Water-content related alterations in macro and micro scale tendon biomechanics. *Sci. Rep.* **9**, 7887 (2019).
6. Chandrasekaran, R. *et al.* Computational design of biopolymer aerogels and predictive modelling of their nanostructure and mechanical behaviour. *Sci. Rep.* **11**, 10198 (2021).
7. Niamnuy, C., Devahastin, S., Soponronnarit, S. & Raghavan, G. S. V. Modeling coupled transport phenomena and mechanical deformation of shrimp during drying in a jet spouted bed dryer. *Chem. Eng. Sci.* **63**, 5503–5512 (2008).
8. Gamble, J. F., Tobyn, M. & Hamey, R. Application of image-based particle size and shape characterization systems in the development of small molecule pharmaceuticals. *J. Pharm. Sci.* **104**, 1563–1574 (2015).
9. Liu, E. J., Cashman, K. V. & Rust, A. C. Optimising shape analysis to quantify volcanic ash morphology. *GeoResJ* **8**, 14–30 (2015).
10. Apinyavisit, K., Nathakaranakule, A., Mittal, G. S. & Soponronnarit, S. Heat and mass transfer properties of longan shrinking from a spherical to an irregular shape during drying. *Biosyst. Eng.* **169**, 11–21 (2018).
11. Panyawong, S. & Devahastin, S. Determination of deformation of a food product undergoing different drying methods and conditions via evolution of a shape factor. *J. Food Eng.* **78**, 151–161 (2007).
12. Yan, Z., Sousa-Gallagher, M. J. & Oliveira, F. A. R. Shrinkage and porosity of banana, pineapple and mango slices during air-drying. *J. Food Eng.* **84**, 430–440 (2008).
13. Devahastin, S., Suvarnakuta, P., Soponronnarit, S. & Mujumdar, A. S. A comparative study of low-pressure superheated steam and vacuum drying of a heat-sensitive material. *Dry. Technol.* **22**, 1845–1867 (2004).
14. Neal, F. B. & Russ, J. C. *Measuring Shape* (CRC Press, 2012).
15. Jinorose, M., Stienkijumpai, A. & Devahastin, S. Use of digital image analysis as a monitoring tool for non-uniform deformation of shrinkable materials during drying. *J. Chem. Eng. Jpn.* **50**, 785–791 (2017).
16. Jahanbakshi, A. & Kheiralipour, K. Evaluation of image processing technique and discriminant analysis methods in postharvest processing of carrot fruit. *Food Sci. Nutr.* **8**, 3346–3352 (2020).
17. Igathinathane, C., Pordesimo, L. O., Columbus, E. P., Batchelor, W. D. & Methuku, S. R. Shape identification and particles size distribution form basic shape parameters using ImageJ. *Comput. Electron. Agr.* **63**, 168–182 (2008).
18. Gil, M., Teruel, E. & Arauzo, I. Analysis of standard sieving method for milled biomass through image processing. Effects of particle shape and size for poplar and corn stover. *Fuel* **116**, 328–340 (2014).
19. Berkman, S. *et al.* The Shape grading system: A classification for growth patterns of pituitary adenomas. *Acta Neurochir.* <https://doi.org/10.1007/s00701-021-04912-1> (2021).
20. Stienkijumpai, A., Jinorose, M. & Devahastin, S. Development and use of three-dimensional image analysis algorithms to evaluate puffing of banana slices undergone combined hot air and microwave drying. In *Proceedings of the 21st International Drying Symposium*, Valencia, Spain, 529–536 (2018).
21. Poonnoy, P., Yodkeaw, P., Sriwai, A., Umongkol, P. & Intamoon, S. Classification of boiled shrimp's shape using image analysis artificial neural network model. *J. Food Process Eng.* **37**, 257–263 (2014).
22. Žunic, J., Kopanja, L. & Fieldsend, J. E. Notes on shape orientation where the standard method does not work. *Pattern Recogn.* **39**, 856–865 (2006).
23. Chakraborty, A. & Sharma, R. A deep crystal structure identification system for X-ray diffraction patterns. *Vis. Comput.* <https://doi.org/10.1007/s00371-021-02165-8> (2021).
24. Lee, I. H. & Chang, K. J. Crystal structure prediction in a continuous representative space. *Comput. Mater. Sci.* **194**, 110436 (2021).
25. Kaufmann, K. *et al.* Crystal symmetry determination in electron diffraction using machine learning. *Science* **367**, 564–568 (2020).
26. Wu, Z. *et al.* Neural network classification of ice-crystal images observed by an airborne cloud imaging probe. *Atmos. Ocean* **58**, 303–315 (2020).
27. Shoji, D., Noguchi, R., Otsuki, S. & Hino, H. Classification of volcanic ash particles using a convolutional neural network and probability. *Sci. Rep.* **8**, 8111 (2018).
28. Wang, Q. & Ward, R. K. A new orientation-adaptive interpolation method. *IEEE Trans. Image Process.* **16**, 889–900 (2007).
29. Lei, S., Osiski, G. O., McIsaac, K. & Wang, D. An automatic methodology for analyzing sorting level of rock particles. *Comput. Geosci.* **120**, 97–104 (2018).
30. The MathWorks, Inc. Pretrained Deep Neural Networks. [https://www.mathworks.com/help/deeplearning/ug/pretrained-convolutional-neural-networks.html?searchHighlight=compared%20pretrained&\\_tid=srchtitle](https://www.mathworks.com/help/deeplearning/ug/pretrained-convolutional-neural-networks.html?searchHighlight=compared%20pretrained&_tid=srchtitle) (2021)
31. The MathWorks, Inc. Regionprops. <https://www.mathworks.com/help/images/ref/regionprops.html> (2020).
32. The MathWorks, Inc. Image Processing Toolbox. <https://www.mathworks.com/help/images/creating-a-gallery-of-transformed-images.html> (2020).
33. Quevedo, R., Carlos, L. G., Aguilera, J. M. & Cadoche, L. Description of food surfaces and microstructural changes using fractal image texture analysis. *J. Food Eng.* **53**, 361–371 (2002).
34. The MathWorks, Inc. trainingOptions. <https://www.mathworks.com/help/deeplearning/ref/trainingoptions.html?sessionid=d360a0f3e9f444dab7e80ef4d2fb> (2021).

## Acknowledgements

The authors express their sincere appreciation to the Thailand Research Fund (TRF) for financially supporting the study through the Grant for New Scholar (Grant number MRG6180302) awarded to author Jinorose and the Senior Research Scholar Grant (Grant number RTA6180008) awarded to author Devahastin. Author Stienkijumpai is grateful to King Mongkut's Institute of Technology Ladkrabang for its financial support.

## Author contributions

P.S.: investigation, formal analysis, writing—original draft preparation. M.J.: conceptualization, funding acquisition, supervision, writing—reviewing and editing. S.D.: funding acquisition, supervision, writing—reviewing and editing.

## Competing interests

The authors declare no competing interests.

### Additional information

**Correspondence** and requests for materials should be addressed to M.J.

**Reprints and permissions information** is available at [www.nature.com/reprints](http://www.nature.com/reprints).

**Publisher's note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits 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 licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence 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. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

© The Author(s) 2021