1947
University of Tabriz

***University of Tabriz***

*Faculty of Electrical and Computer Engineering*

*Department of Computer Engineering*

Bachelor's Thesis

in Computer Engineering

Thesis Subject:

# Froth Flotation by Using Computer Vision Algorithms

Supervisor:

**Dr. Abdolhamid Moallemi Khiavi**

Researcher:

**Sina Razi Moftakhar**

**Winter 2022**

## Chemical Background:

**Purification process using flotation:**
There are numerous ways in the industry so as to purify chemical materials. Although expenditures of purification in the economic calculation are very significant, we have to take into account some factors in order to mitigate this procedure. These factors are given below:

- characteristics of the materials that are going to be purified
- characteristics of impure materials
- operating processes of purification
- purification costs and budget

One of the industrial and low-price ways of purification of materials is froth flotation. This approach is used in minerals, inorganics, and materials which themselves and their impurity show electrical properties.
Flotation is divided into two main categories, which are direct flotation and indirect flotation. According to the direct method, the main product is separated from impurities in the form of froth, and sludge remains. In terms of the indirect method, impurities are separated along with froth, and the main product is collected and separated in the form of sedimentary slurry.

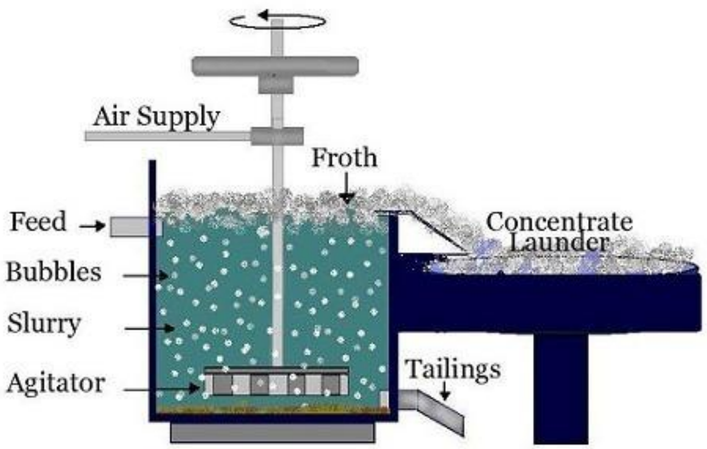


Figure 1: Flotation Process

**How is the flotation process performed?**
All of the procedures of the flotation process are designed and performed in terms of the primary properties of the main product and its impurities. All of the steps are indicated below:

1. Raw materials should be ground and powdered by crushers and mills. The finer the raw material is ground, the better results are gotten.
2. Raw powdered material should be poured to specific tanks with an agitator, and should be mixed with water by determining of water flow rate and amount of raw material.

3. Adding specific chemical materials, such as collectors, regulators, and froth generators.
4. Blowing of air by roots-type blowers, and controlling of air flow rate.
5. Entry of material to the low-depth pool or tank with rotary mechanical paddles. These rotary paddles separate generated froth by considering elements, such as concentration, strength or stability, shape, and size of bubbles.

The flotation process considers the design of bubbles through chemical additives, and connect them to either the main product or impurities, and make them light and float.

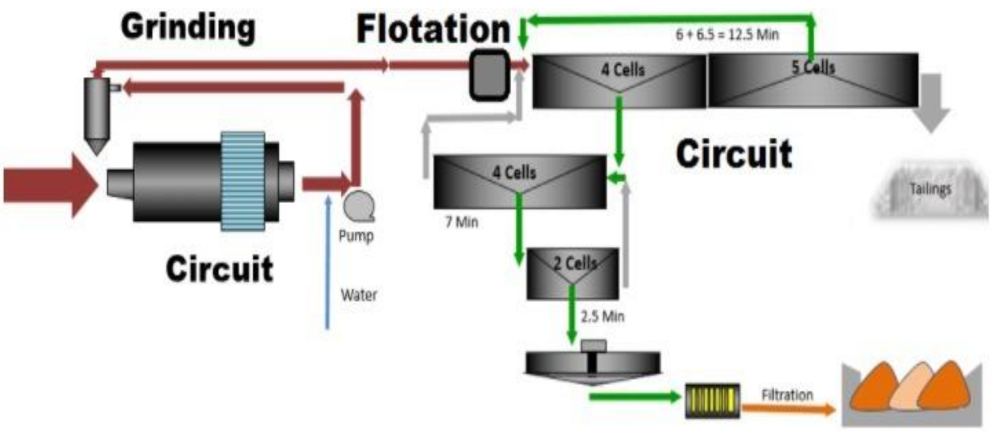


Figure 2: Flotation Process's ingredients

In direct flotation, blowing air and additives are attached to the main product. After being light regarding to weight, they are brought to the surface of the pool or tank along with froth. Next, impurities will convert to a sediment as sludge, and will be pushed into the drainage. Finally, the main product will be collected by a paddle which will send the product to the next step.

Regarding to indirect method, in spite of the steps mentioned in the previous method, blowing air and additives are attached to impurities, being sent out by paddle in the form of froth. In the following step, the main product will convert to a sediment. Finally, it will be collected and be sent to the following step.

In the flotation process, the main product and its impurities should be different from each other in terms of either hydrophilic properties or the chemical additives which trigger this condition. In this process, either the main product or its impurities have to be a hydrophilic substance, and the other one has to be a hydrophobic substance.

**Chemical materials and additives of flotation process:**
In this process, there are a considerable number of chemical materials and additives which are used in the forms of chemical feeders. For instance, collectors, froth generators, and regulators.

1. Collectors: they are added to minerals so as to cause hydrophobic features.
2. Froth generators: they have a significant impact on generating froth stability, and they do not let bubbles merge.
3. Regulators: they are widely used in increasing effects of collectors which were added to minerals. These are divided into three groups which are activators, depressants, and dispersants.

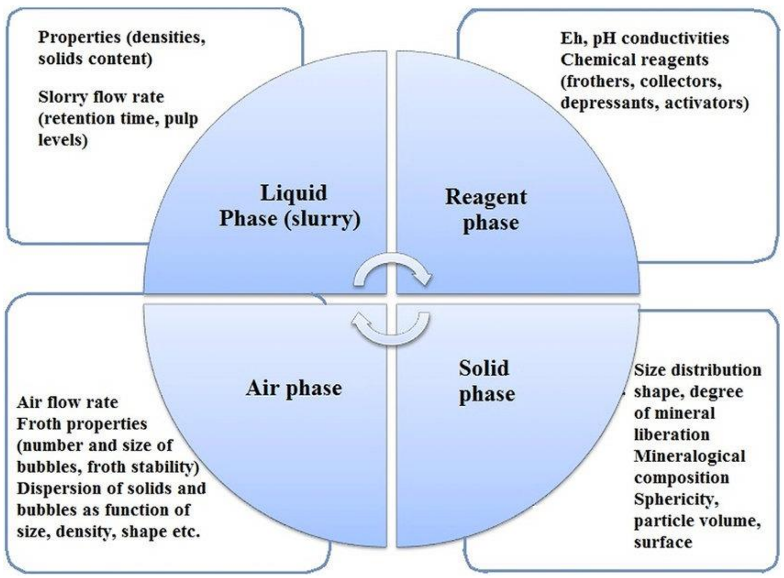


Figure 3: Four main phases of Flotation process

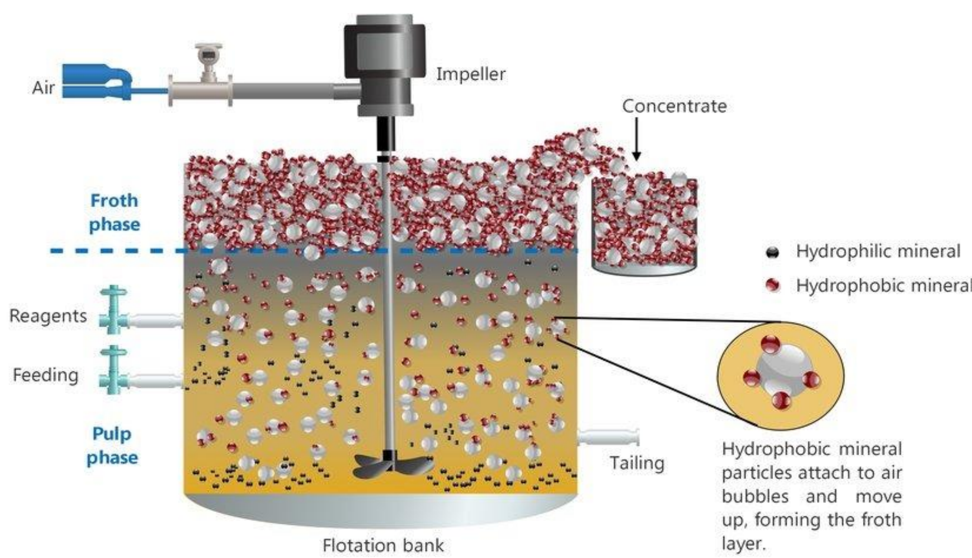


Figure 4: Flotation bank

**Control operation of the flotation process:**

This process corresponds directly with the experience of the operator. The operators who have experienced a substantial number of things in their works are able to control this process based on their practical knowledge and the previous operators' experiences. The results of laboratory and purity of the main product demonstrates the accuracy of the operations. Furthermore, this term can be used when it comes to chemical additives.

Operator who works in this unit should take into account some factors, which were mentioned above, such as concentration, strength or stability, shape, and size of bubbles. Moreover, he can comfortably determine with the wisdom of hindsight whether to take out the generated froth or not based on determined time. The operator is designated to control this process personally, and the accuracy of one operator differs from the other one due to their skills and experience. Therefore, the final purity of the final product will not as same as each other.

Figure 5: An operator who has to determine the proper time of the process by looking into the bubbles, and decide whether to do something or not.

Figure 6: Mechanical arm in order to collect the main product

In the automatic control of this process, all of the inlets, namely crushing step, mills, sizing, flow rate of the raw material and water, and chemical additives, are controlled by instruments which are analyzers, transmitters and PLC. All of these instruments have been ubiquitous nowadays, and they can be provided without any difficulty. However, control of the bubbles is quite arduous due to the fact that it requires hours of programming.

According to the artificial intelligence, first of all, special illustrators in the form of clusters cast light on the froth. Then, special video recorders are provided in order to snap photos of these frames. In final, all of the AI-based steps will be discussed in this thesis.

**Cluster shaped illustrators:**

There are dozens of lights which are put together in a circular shape or it would be better to say, in the form of a cluster. They cast light to surface of chemical material, and some projected shadows will be generated on top of them. The density of specific metal, such as cooper, is lower than the density of mixed water because cooper is along with the bubbles on top of the tank. As a result, the cooper is separated by means of froth. The bubbles contain the specific metal, and if these bubbles burst, the metal will not remain in the surface of tank. The brighter the shadows seem, the lower density the bubbles have. Therefore, the bubbles will be exposure to bursting, and losing chance of collecting the main product. The important factor is that bigger bubbles could be burst easily, and smaller ones could be merged together. The medium sized bubbles are the best ones so as to collect the main product.

## The hypothesis:

One of the biggest issues which could be triggered by operators is that do not keep track of the proper time of the process. They have to consider the process and control it when it comes inflating or bursting the bubbles. Whenever they witness any change in the process, they have to make a right decision in order to follow appropriate procedure. However, there are numerous reasons why human cannot control this system in a proper way, such as differences in their experience, exhausting of heavy workload which may decrease their abilities. Therefore, it would be recommended to provide computer vision based system so as to reduce these kinds of inaccuracies.

## The problem:

This intelligent system should virtually determine the position of shadows which will be generated by cluster shaped illustrators on top of the chemical tank. Finally, the results should be collected in order to make a correct decision when it comes to any change in bubbles.

## Dataset:

One of the main prerequisites of the learning procedure of the AI-based systems is the dataset. We have to prepare appropriate dataset based on our problem in order to train better with higher accuracy. If we do not do so, we will fail in terms of accuracy, and the problem will remain unsolved. In this problem, fortunately, I see one of the best datasets that I have ever seen due to the reasons which will be shown below:

a) Number: one of the main elements in the training step is the number of dataset. The more data we train, the higher accuracy we get.
b) Similarity: each datum should be similar to another one because we just want to solve a specific problem
c) Size: one of the basic steps of deep learning is convolutional neural networks (CNN). In order to train with these kinds of algorithms, we have to add data with same size (same matrix). If this important step is not satisfied, the algorithm will not work, and there will be no result.

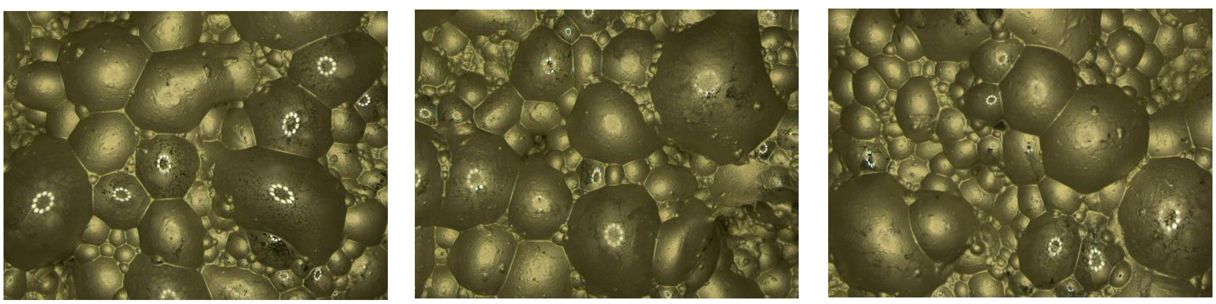

Figure 7: The given dataset.

Despite the pros mentioned above, I should declare some obstacles that I had to mitigate regarding to the dataset.

a) Format: the format of given dataset was '.tiff'. I noticed that the CNN algorithm did not work with given dataset. So as to convert the dataset to another format, such as 'jpg', I wrote a code with open-cv library. However, it did not work because it just had changed the name of the files. Therefore, I made a decision to convert my data with one of the convertor websites, and finally it worked properly.
b) Size: although these data are clear and high-quality, my system is not very powerful in order to train these size of dataset. So, I had to resize the data to smaller size from '1440*1080' to '256*256'.
c) Number: while mostly large number of data can help algorithm to train with higher and better accuracy, my system is not very expensive and powerful to do so. Therefore, I had to use only small number of them, and use pre-trained models such as image-net which contains millions of trained photos.

d) Hardware: as mentioned previously, although my CPU is quite suitable which is ‘i7 core’, my GPU is not very powerful. In the training, if we train with the CPU, the memory will be overflow, and it will damage the laptop. As a result, it would be better to train with the GPU which can comfortably overcome the arithmetic calculations, namely matrices, parallel calculations, etc. As a matter of fact, my GPU is not quite appropriate to this process, and I have to use online cloud services, such as ‘Google Colab’. The more accurate results we want to get, the more money should be paid to service providers in order to let us use their hardware resources.

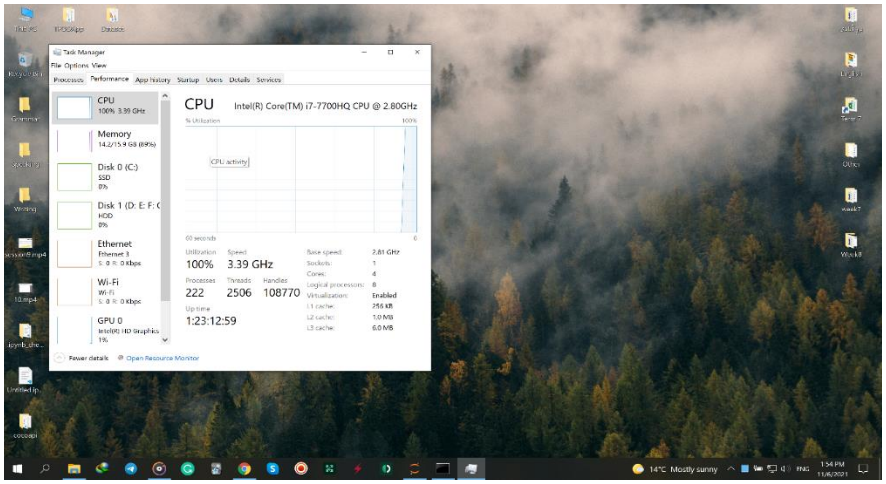


Figure 8: My CPU which was using its 100% memory while I was training the model.

## Algorithms:

There are a considerable number of algorithms that can tackle this problem. These algorithms should detect and count the nucleus (cooper of the froth) in an excellent way. As a result, in this thesis, two state-of-the-art approaches will be discussed.

- Object detection
- Instance segmentation

Every approach has its pros and cons, and none of them outweighs the other one. For instance, object detection is faster than instance segmentation due to its simpler architecture, while instance segmentation is more accurate than the other one.

## Object detection:

**History:** object detection is a computer technology that deals with detecting instances of semantic objects of a special class (e.g. humans, buildings, or cars) in digital images and videos. It is related to computer vision and image processing. It considers the problem as two approaches rather than one approach, which is in other deep learning algorithms. The outputs should determine two solutions which are bounding box and classified answer, to say the least. Sample boxes from the image have to apply the image classifier, and non-background boxes are detections (figure 9).

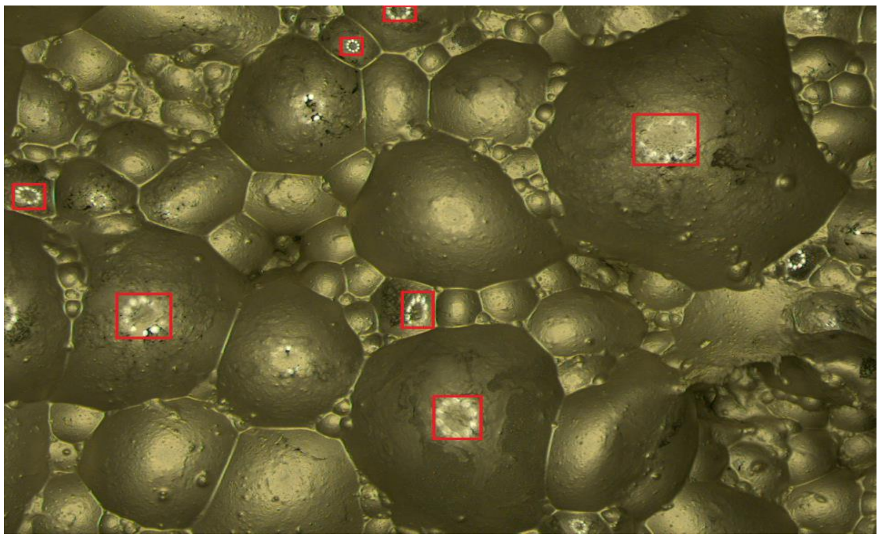

Figure 9: Object detected froth flotation dataset through object detection algorithms

There are many ways to sample boxes, such as sliding windows and region proposals. The sliding window is very expensive due to time, and it has been made obsolete. Therefore, in most computer vision architectures, region proposal is used by developers and scientists. One of the primary algorithms that have been released is R-CNN (Girshick, 2013) which did object detection in two steps. The first step is classifying regions by bounding-box regression, and the second one is classifying objects with SVM (M.A. Hearst, 1998), which requires hours of training, and it has three independently trained components (not end-to-end) (figure 10).

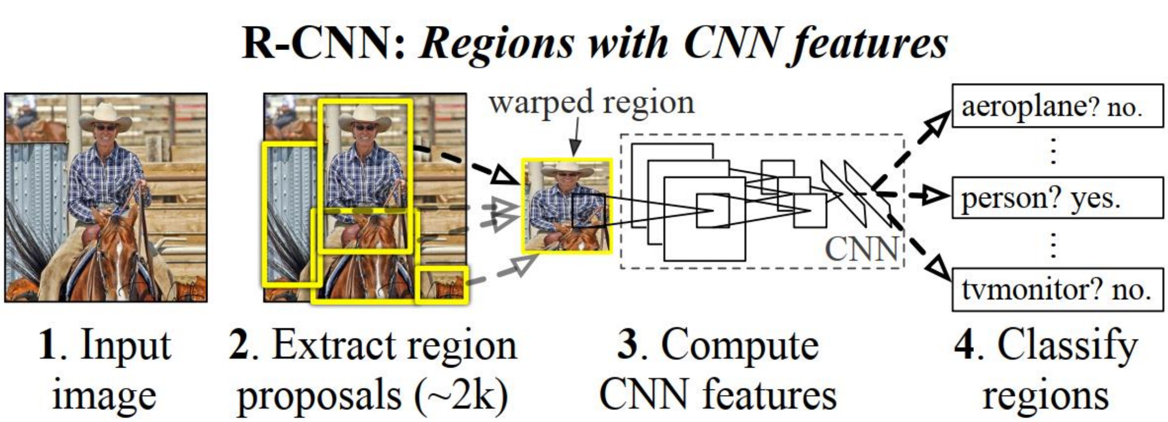


Figure 10: R-CNN: Regions with CNN features (Girshick, 2013).

Although there are plenty of cutting-edge architectures, such as YOLO (you only look once) and Faster R-CNN (end-to-end version of R-CNN), I decided to use an alternative whose name is Single Shot Multi-Boxes Detector which is known as SSD (Jeong-Seon Lim, 2019). This architecture takes images and processes them in three steps, known as feature extraction, detection heads and NMS, respectively (figure 11).


Figure 11: Three main steps of SSD architecture

**SSD:**

As mentioned previously, this approach has three steps. The first step, which is called feature extraction, encodes features at different scales. The algorithm should take the image which a specific matrix of pixels, and encode them via specific architecture, namely VGG, Inception-net, and Res-net. I made a decision to use Deep Residual Network (Kaiming He, 2015), to be more precise deep res-net 50 (figure 12), even though there are numerous versions of this architecture, such as res-net 34, res-net 50 and res-net 101. The imperative reason to do so is that res-net 50 has lower layers than the other versions which can finish the encoding process as fast as possible in a plausible accuracy.

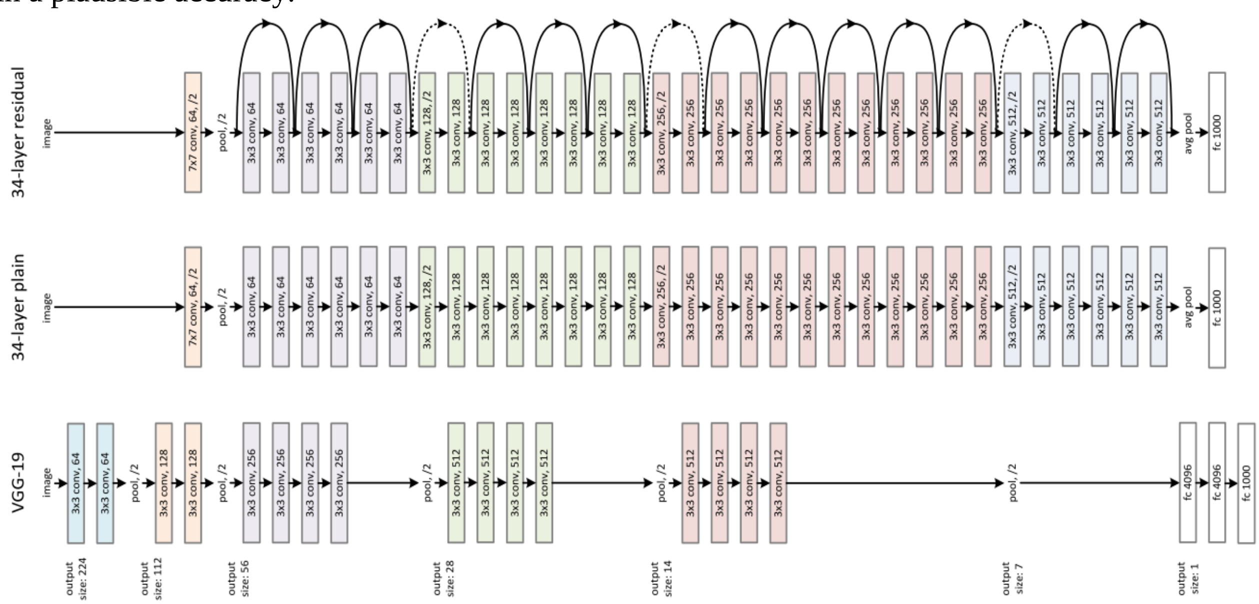


Figure 13: Comparing three CNN architectures (Kaiming He, 2015).

If we take a look at the above figure, it is obvious that Res-net architecture is superior to the others due to the fact that it has shortcuts that increase the pace of the training. Whenever neurons get caught up with heavy arithmetic calculations which will increase the training time, shortcuts would help the check-pointed neurons to jump to another layer so as to finish the training as soon as possible.

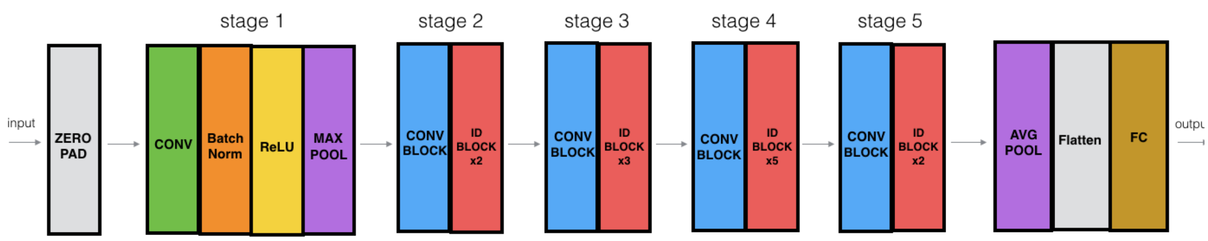


Figure 12: Deep residual network with 50 layers. Each block, such as identity block, consists of multi layers. For example, batch normalization and 2D convolutional.

The second step calls detection heads (figure 14), taking feature maps of the first step which were prepared by res-net 50 and extra feature layers (it differs from one problem to another one), generates boxes and score classes. Before that, we should take into account two things. Multiple layers handle different scales, and different filters predict boxes of different shapes/ sizes.

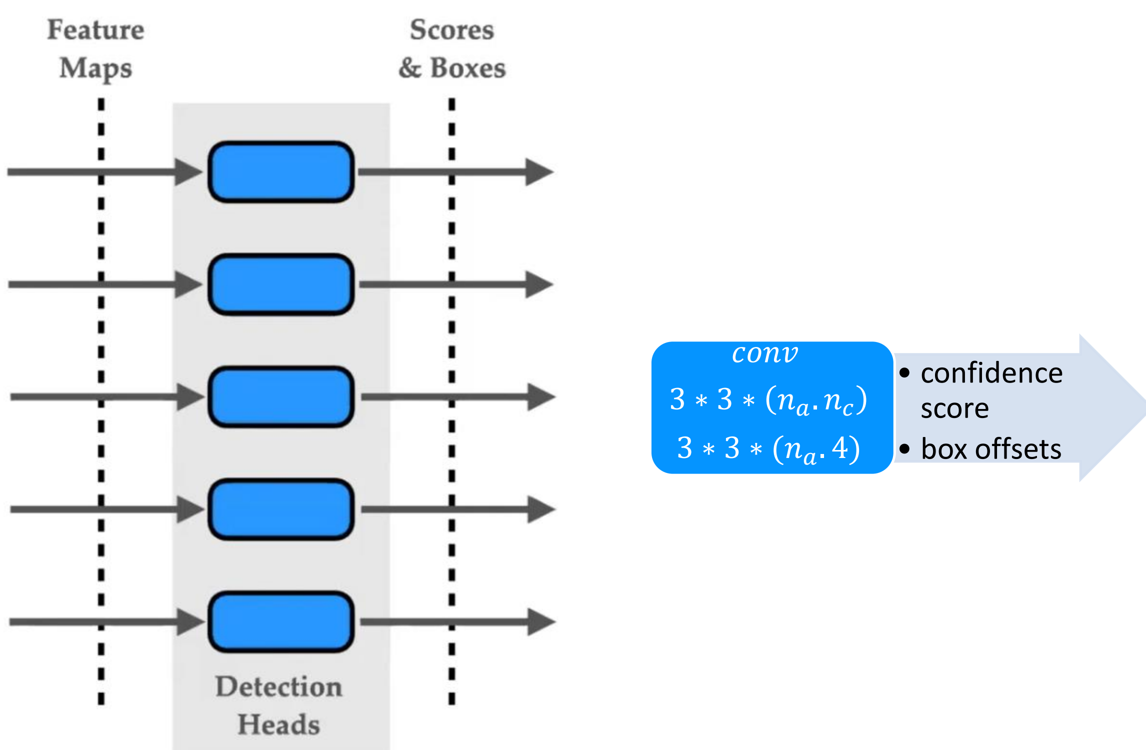


Figure 14: SSD architecture’s Multi-box detector

Based on the 14th figure, it is clear that detection heads convert feature maps to scores and boxes. Each head carries out special task which are calculating confidence score and box prediction. While confidence scores are calculating background and tick, box predictions are performing calculation of NN outputs, default box and predictions (figure 15).

NN Output: $(\beta_x, \beta_y, \beta_w, \beta_h)$

Default box: $(d_x, d_y, d_w, d_h)$

Prediction: $(b_x, b_y, b_w, b_h)$

$$b_x = d_x + d_w\beta_x \qquad b_w = d_w e^{\beta_w}$$

$$b_y = d_y + d_h\beta_y \qquad b_h = d_h e^{\beta_h}$$

Figure 15: Box prediction’s outputs and configurations

Labels: before each step, we should know how to label our algorithm. In object detection problems, the labels are divided into two main categories. Positive labels

are kinds of labels which consider class target and offset target for default box d. However, negative ones are kinds of labels which their default box containing no object. In addition, in positive ones, we have to calculate intersection over union (IOU) (figure 16), and if it is larger than specific threshold (usually 0.5), then it will be correct.

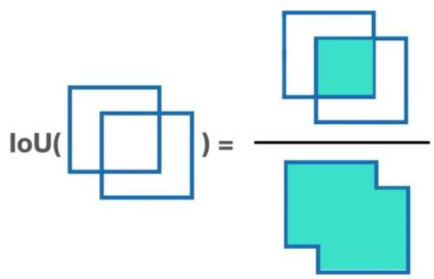


Figure 16: intersection over union (IOU)

Loss: the loss function is a function that measures the discrepancy between the algorithm's actual output and the predicted output. It's a means of evaluating how well your algorithm models the data. The network output determines two things which are (logit for class c) and (offset target) for default box d. In the given figure, all of the necessary measurements are shown (figure 17).

$$L_{loc} = \sum_{(\hat{c}_d, \hat{\beta}_d) \in Pos} \ell_{loc}(\beta_d - \hat{\beta}_d)$$

Typically $\ell_{loc}$ : smooth $\ell_1$

$$L_{conf} = -\sum_{(\hat{c}_d, \hat{\beta}_d) \in Pos} \log(\gamma_{d,\hat{c}_d}) - \sum_{d \in Neg} \log(\gamma_{d,bg}) \qquad \gamma_{d,c} = \frac{\exp(\sigma_{c,d})}{\sum_{c'} \exp(\sigma_{c',d})}$$

(probability for class c)

Figure 17: All of the necessary measurement of SSD model.

The 3rd step is non maximum suppression (NMS) (Jan Hosang, 2017). Candidate regions for the object of interest are referred to as proposals. The majority of methods use a sliding window over the feature map to assign foreground/background scores based on the features calculated in that window. The scores of the neighborhood windows are equal to some extent, and they are regarded prospective areas. Hundreds of suggestions result as a result of this. We preserve loose constraints in this step since the proposal generating process should have a high recall. However, it is time consuming to analyses all of these ideas through the classification network. This leads to a technique known as Non-Maximum Suppression, which filters suggestions depending on a set of criteria.

To say the least, this algorithm removes duplicate proposals and retain proposals which have higher IOU. Firstly, it selects proposals with highest confidence scores. Then the proposals with calculated IOU are being calculated, and redundant ones are removed. Then, this procedure is repeated in the remaining proposals. In following step, after calculating IOU of proposals, the boxes with higher IOU than threshold will be eliminated. Finally, this step will continue until there will be no redundant proposal.

Finally, after all of these three steps have been processed, the algorithm predicts the model. In modified versions of SSD, the new Feature Pyramid Network (FPN) layer is added. It substitutes the feature extractor in detectors, and provides multiple feature map layers (multi-scale feature maps) with higher quality information for object detection rather than the normal feature pyramid (figure 18).

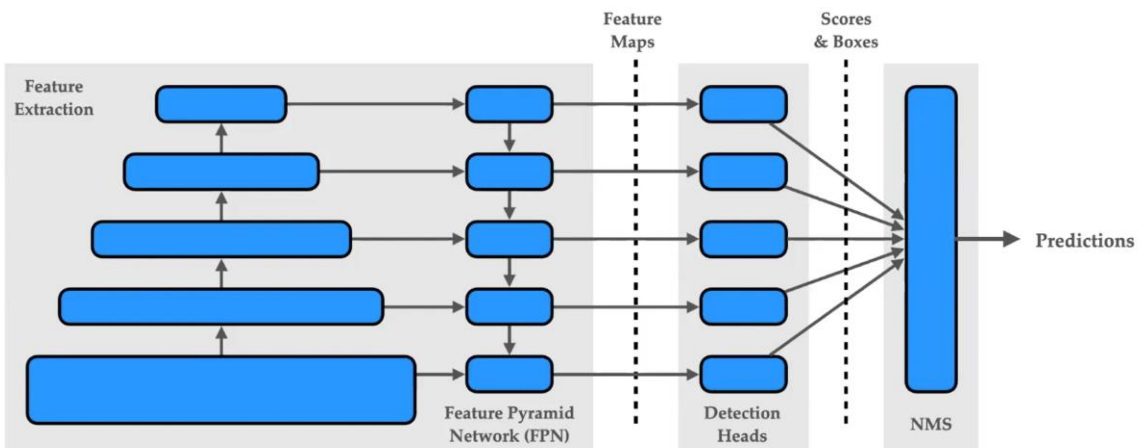


Figure 18: modified architecture of SSD.

The details of this approach will be analyzed in the following sections.

**Instance segmentation:** History: before discussing about instance segmentation, firstly, it would be better to talk about an intrinsic object detection method, called Faster R-CNN (Shaoqing Ren, 2016).

**Faster R-CNN:** this architecture is improved version of R-CNN with two main aspects which are Region Proposal Network (RPN) and Fast R-CNN. RPN is a Neural Network that computes multiple objects that are prevail within a specific image. According to Fast R-CNN, it is used in feature extraction by using RoI-Pool (Region of Interest Pooling) of each candidate box, and it accomplishes bounding box regression and classification. RoIPool is used in extracting a small feature map from every RoI when it comes to detection (figure 19).

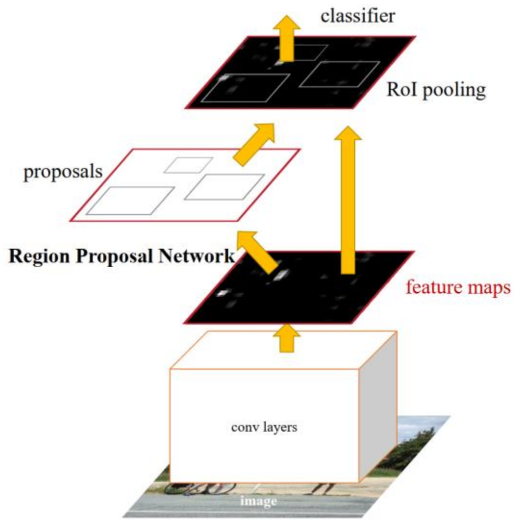


Figure 19: Faster R-CNN is a single, unified network for object detection. The RPN module serves as the 'attention' of this unified network (Shaoqing Ren, 2016).

Although there are a substantial numbers of segmentation algorithms, such as semantic segmentation, which can be used in image segmentation, froth flotation is a quite difficult problem which requires precise computation. In the froth flotation, every instance of the detected objects has certain meaning, and it differs from other objects (projected shadows of cluster shaped illustrators). It needs an algorithm which can easily distinguish between different objects, and considers them as unique objects. In this thesis, a new state-of-the-art approach, known as instance segmentation, to be more exact, Mask R-CNN (Kaiming He, 2018), will be covered.

**Mask R-CNN:**

Mask R-CNN is a CNN that is the state-of-the-art in image and instance segmentation. Faster R-CNN, a Region-Based Convolutional Neural Network, has been used to create Mask R-CNN. Understanding the notion of Image Segmentation is the first step in understanding how Mask R-CNN works. The problem of computer vision is the process of segmenting a digital picture into many segments is known as image segmentation (sets of pixels, also known as image objects). Objects and boundaries are located using this segmentation (lines, curves, etc.).

Instance Segmentation, also known as Instance Recognition, focuses on accurately detecting all objects in a picture as well as segmenting each instance. As a result, it is a mix of object detection, localization, and classification. To put it another way, this sort of segmentation takes a step further by clearly distinguishing each object classified as a similar instance.

Faster R-CNN has been used to create Mask R-CNN. While Faster R-CNN gives a class label and a bounding-box offset for each candidate object, Mask R-CNN adds a third branch that outputs the object mask. The additional mask output differs from the class and box outputs in which it requires a much finer spatial layout of an item to be extracted.

Mask R-CNN is a Faster R-CNN modification which works by adding a branch for predicting an object mask (Region of Interest) alongside the current branch for bounding box recognition (figure 20).

This algorithm has numerous benefits like simplicity, performance, efficiency (adds only a negligible amount of overhead) and flexibility.

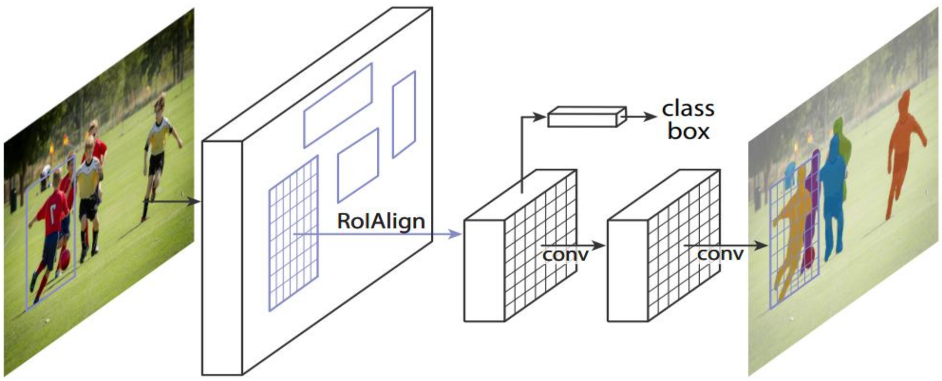


Figure 20: The Mask R-CNN framework for instance segmentation (Kaiming He, 2018).

Before diving into the Mask R-CNN profoundly, based on 20th figure, Fully Convolutional Networks (FCN) (Xiaolong Liu, 2018) should be implemented on the last CONV boxes (FCN on RoI) in order to generate pixel wise prediction (segmentation). The architecture of FCN is indicated below(figure21).

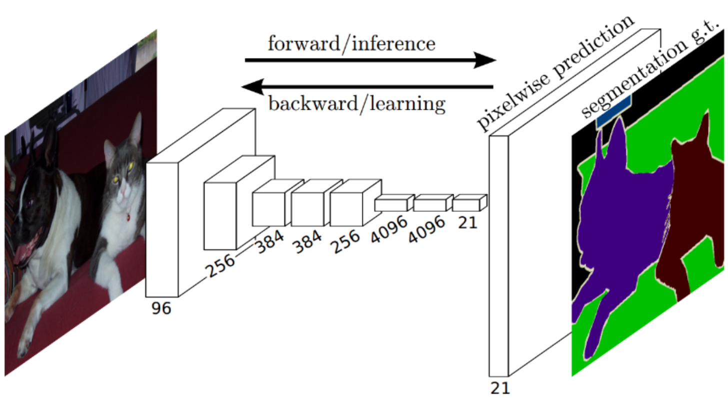


Figure 21: Fully convolutional networks can efficiently learn to make dense predictions for per-pixel tasks like semantic segmentation (Xiaolong Liu, 2018).

Basic architecture of Mask R-CNN consists of RPN, RoI Align and parallel prediction for the classes, box and binary mask for each RoI. Segmentation differs from most prior systems in which classificatoin corresponds with mask prediction. The loss function which determines the discrepancy of the predicted value and real value is computed by sum of three factors which are class loss, box loss and mask loss (figure 22). Loss for classification and box regression is as same as Faster R-CNN. A per pixel sigmoid is applied to each map, and average binary cross entropy loss (Katarzyna Janocha, 2017) (figure 23) is defined to the map loss. Mask loss is only defined to the ground truth class and class prediction and mask generation are decoupled.

$$H_p(q) = -\frac{1}{N}\sum_{i=1}^{N} y_i \cdot log(p(y_i)) + (1 - y_i) \cdot log(1 - p(y_i))$$

Figure 23: Binary Cross-Entropy / Log Loss

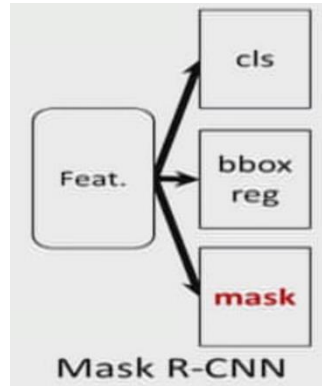


Figure 22: The Mask R-CNN's three main part.

By considering the 20th figure, one of the significant units of the Mask R-CNN is RoI align. The main motivation of this part is not using quantization for data pooling. Firstly, the image as input is taken by Mask R-CNN. After being processed by CNN backbone (various architectures, such as VGG, Res-net, etc., can be used.), it is quantized, and it will be sent to RoI align unit. It removes this quantization which triggers this misalignment. For each bin, 4 locations are sampled, and bilinear interpolation is done (a resampling approach that estimates a new pixel value using the distance weighted average of the four nearest pixel values). The number of samples or the specific location of sampling has no effect on the results (figure 24).

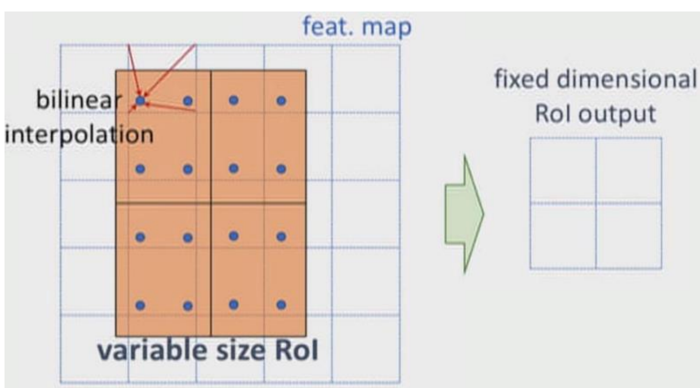


Figure 24: The architecture of RoI align.

It is possible for the network to be divided into two parts which are backbone architecture and network head. Backbone is used for feature extraction, and network heads include object detection and segmentation components. Network heads have almost the same architecture as Faster R-CNN, except with the addition of a convolution mask prediction branch.

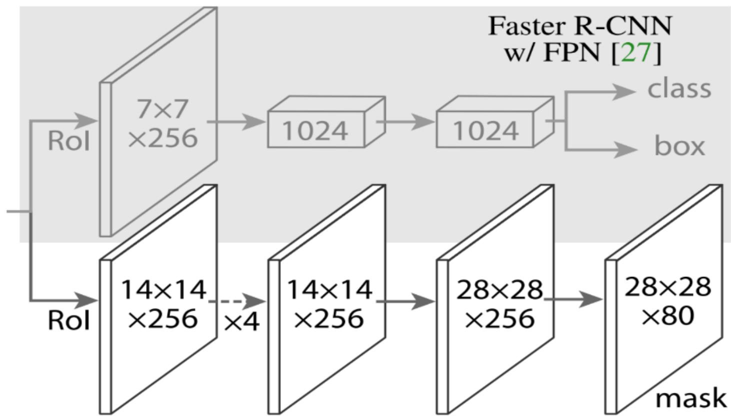


Figure 25: The precise architecture of Mask R-CNN, which indicates Faster R-CNN, which generates bounding box and classified answer, and FCN Mask head.

## Experiments:

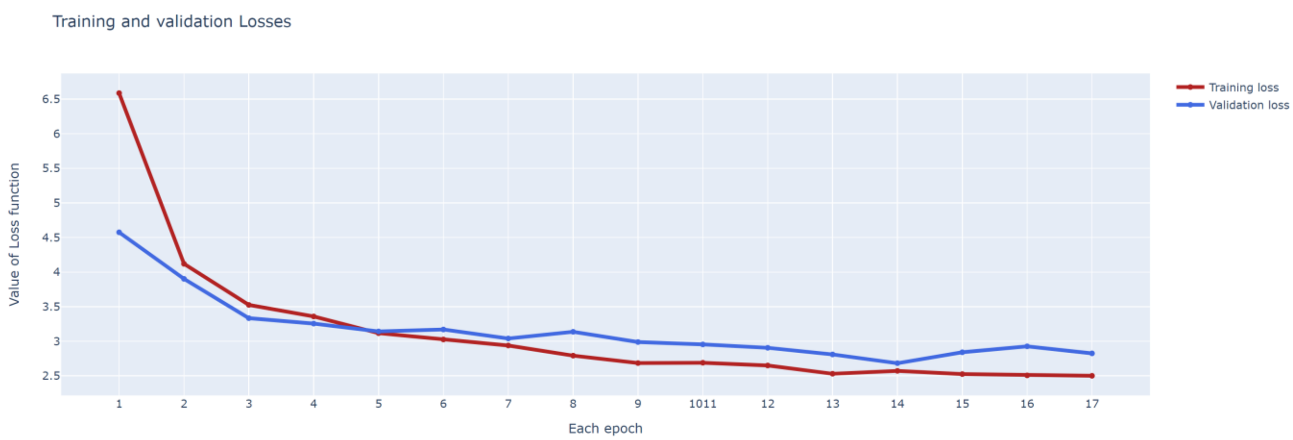


Figure 26: The loss plot of trained and validated SSD on froth flotation dataset.

I annotated more than a thousand of the froth flotation dataset by ‘labelimg’, and then converted to a single file using pickle to a ‘.p’ file so as to train the whole annotated dataset easier. Unfortunately, I was unable to train the whole dataset due to the fact that will be mentioned. Not only, based on 27th figure, each epoch so as to be trained, took more than half an hour but also I had to train my model on ‘Google colab’ which always halts on specific period of time. Therefore, I could only train 500 of the froth flotation dataset with merely 17 epochs because ‘Goole colab’ did not let me train more. Finally, I could get normal mean average precision accuracy, to be more precise recall/ precision on a ROC curve. If I could train more data with more epochs, I got even better accuracy. In this problem, by these obstacles, I got at least 60% (from 60% to 80%). Based on the differences of each image, this value can comfortably rise to

90%.  Object detection is really difficult problem, and in order to train it, scientists usually use image-net dataset which comprises more than a million data. However, I was not able to use it since the froth flotation dataset is a unique dataset that transfer learning cannot improve its accuracy. In my opinion, accuracy of more than 60%, and validation loss of less than 3 (figure 26) is fabulous in this problem because the labeling process is quite difficult since it requires the factory's operators experience so as to be annotated better.

```
Epoch 2/30
368/374 [==========================>.] - ETA: 26s - loss: 4.1246Epoch 00001: saving model to /content/checkpoints/weights.01-3.90.h5
384/374 [============================] - 1749s - loss: 4.1190 - val_loss: 3.9018
Epoch 3/30
368/374 [==========================>.] - ETA: 26s - loss: 3.5236Epoch 00002: saving model to /content/checkpoints/weights.02-3.33.h5
384/374 [============================] - 1726s - loss: 3.5252 - val_loss: 3.3334
Epoch 4/30
368/374 [==========================>.] - ETA: 26s - loss: 3.3639Epoch 00003: saving model to /content/checkpoints/weights.03-3.25.h5
384/374 [============================] - 1715s - loss: 3.3581 - val_loss: 3.2550
Epoch 5/30
368/374 [==========================>.] - ETA: 25s - loss: 3.1305Epoch 00004: saving model to /content/checkpoints/weights.04-3.14.h5
384/374 [============================] - 1699s - loss: 3.1167 - val_loss: 3.1418
```

Figure 27: Each epoch which took at least 30 minutes to be trained.

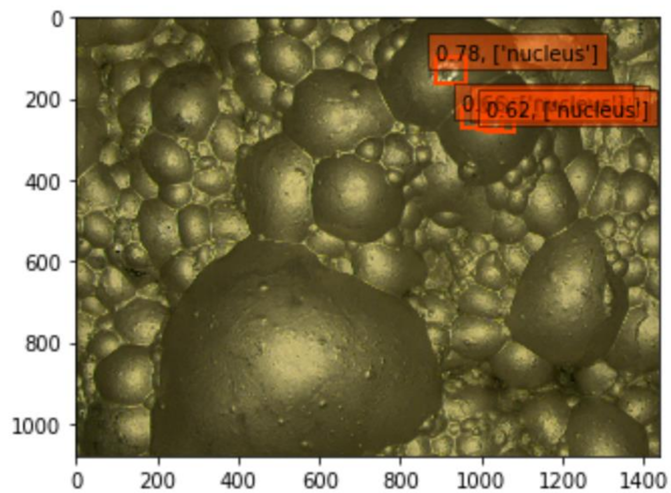


Figure 28: Predicted dataset of the forth flotation process.

The various losses of the dataset on Mask R-CNN determine that the algorithm works properly due to the fact that losses fit perfectly, and virtually there is no neither under-fitting nor over-fitting. These probable problems significantly reduce the final test and validation accuracies. The primary reasons for them are the weakness of either architecture or lack of dataset; however, in this problem, there were no such deficiencies. In addition, validation losses, which determine the correctness of the algorithm on the dataset, indicating that the Mask R-CNN works well on the dataset as well as high accuracy in the detection step (figure 26).

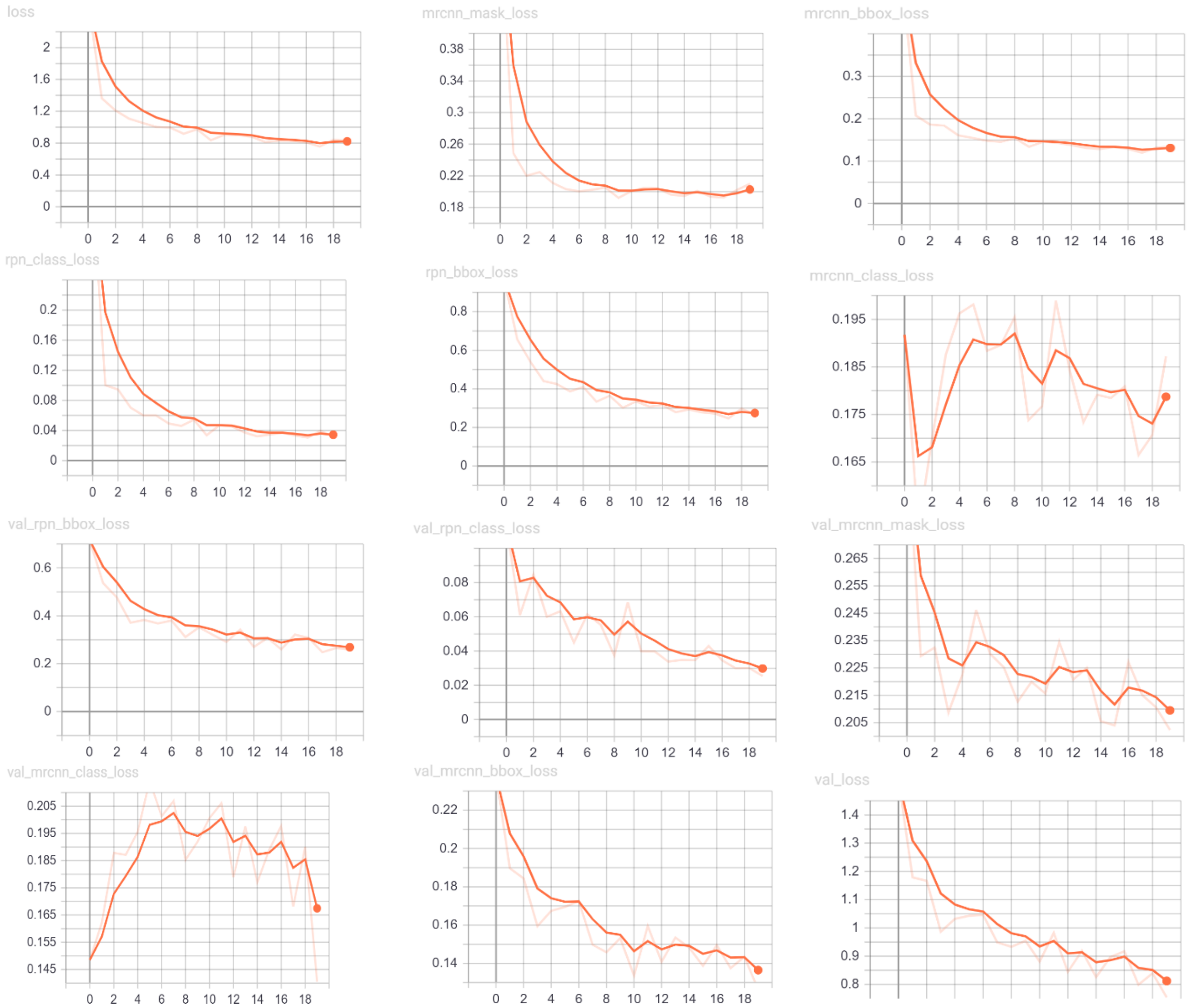


Figure 29: The training losses and validation losses of region proposal network (RPN)'s bounding box and class as well as Mask R-CNN's class, bounding box and mask.

# Conclusion:

In this thesis, two state-of-the-art computer vision models were scrutinized, and each model had specific advantages and disadvantages. According to the real-life froth flotation problem, certain architecture should be chosen and implemented. If the pace of the algorithm is quite necessary, it makes sense to choose SSD-RESNET because it is faster than the other one when it comes to real-time detection. Furthermore, it has simpler architecture (there is no mask and additional overhead) which is very useful in real-time industrial projects. On the other hand, froth flotation is quite a sensitive and challenging project, and every projected shadow has a specific meaning. Mask R-CNN is an architecture that can easily distinguish between instances of objects, and classify them with segmentation and object detection methods along with more precise accuracies. As a result, our detected objects are not similar to each other; to be more exact, they are unique, and this term can help the expert system to decide in a better way. Better decisions will increase the efficiency of the expert system, and help the company to manage their time and resources properly since it can help them stand out from other companies.

## References:


i. *A review: Comparison of performance metrics of.* **al, S A Sanchez et. 2020.** 2020. IOP Conference Series: Materials Science and Engineering.

ii. **Girshick, Ross. 2013.** *Rich feature hierarchies for accurate object detection and semantic segmentation.* s.l. : UC Berkeley, 2013.

iii. **Jan Hosang, Rodrigo Benenson, Rodrigo Benenson. 2017.** *Learning non-maximum suppression.* Saarbrücken, Germany : arXiv, 2017.

iv. **Jeong-Seon Lim, Marcella Astrid, Hyun-Jin Yoon, Hyun-Jin Yoon. 2019.** *Small Object Detection using Context and Attention.* s.l. : arXiv, 2019.

v. **Jonathan Long, Evan Shelhamer, Trevor Darrell. 2015.** *Fully Convolutional Networks for Semantic Segmentation.* UC Berkeley : arXiv, 2015.

vi. **Kaiming He, Georgia Gkioxari, Piotr Dollar. Ross Girshick. 2018.** *Mask R-CNN.* s.l. : arXiv, 2018.

vii. **Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun. 2015.** *Deep Residual Learning for Image Recognition.* s.l. : arXiv, 2015.

viii. **Katarzyna Janocha, Wojciech Marian Czarnecki. 2017.** *On Loss Functions for Deep Neural Networks.* s.l. : arXiv, 2017.

ix. **M.A. Hearst. 1998.** *Support vector machines.* s.l. : IEEE, 1998.

x. **Shaoqing Ren, Kaiming He, Ross Girshick, Jian Sun. 2016.** *Faster R-CNN: Towards Real-Time Object.* s.l. : arXiv, 2016.

xi. **Xiaolong Liu, Zhidong Deng, Yuhan Yang. 2018.** *Recent progress in semantic image segmentation.* s.l. : arXiv, 2018.