KU LEUVEN

FACULTY OF ENGINEERING TECHNOLOGY

TECHNOLOGY CAMPUS DE NAYER

# Counting shells

## Are current neural networks performant enough to count various types of shells in an uncontrolled environment

**Tijs VAN KAMPEN**

Promotor: Prof. dr. Toon Goedemé

Masterproef ingediend tot het behalen van de graad van master of Science in de industriële wetenschappen: E-ICT Software Engineer

Academic year 2022 - 2023

# Summary

Every year the Flemisch Institute for the sea organises a shell counting day to map the diversity of our seaside. Thousands of volunteers go to the beach to count and classify shells. In this thesis we will try to automate this process by using neural networks. The goal is to be able to count the shells in an uncontrolled environment so the volunteers would only have to take pictures of the shells and the neural network would do the rest.

Of course this is not a trivial task, as the shells are not always in the same position, the lighting conditions are not always the same and the shells are not always of the same size.

# Abstract

Het extended abstract of de wetenschappelijke samenvatting wordt in het Engels geschreven en bevat 500 tot 1.500 woorden. Dit abstract moet niet in KU Loket opgeladen worden (vanwege de beperkte beschikbare ruimte daar).

**Keywords**: Voeg een vijftal keywords in (bv: Latex-template, thesis, lang document, ...)

# Contents

# Chapter 1

# Introduction

Every year, one day per year, nearly a thousand volunteers travel to the Belgian coast to collect and categorize the shells that wash up on the beach. This data is collected by the flemish institute for the sea (VLIZ) to study populations of marine molluscs and the impact of their environment (climate change, fishing, etc) on the population. The volunteers participating in this study are mostly enthusiasts, but also scientists and families with children. To ensure a good quality of the data, most volunteers participate in a workshop. The counting of the shells is done by walking along the beach and noting every shell that is found individually. This is a very time consuming process, and the volunteers are often not very experienced in counting, resulting in mistakes with all but the most common shells. When a volunteer finds a shell that they are not familiar with, they can contact a helpdesk to help them. The flowchart of the current process can be found in figure 1.1. Marked blue are the volunteer's actions, marked red are the parts that involve experts.

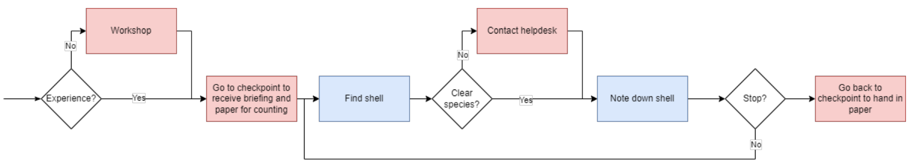


**Figure 1.1:** The current process of collecting data.

The fact that the project relies on volunteers to do most of the legwork, combined with the fact that experts have to man the checkpoints and the helpdesk, makes the project unscalable beyond having a single dedicated day per year. With over 5 million people visiting the Belgian coast every year, there is a lot of potential to collect more data if the process of collecting the data could be simplified to be accessible to anyone visiting the beach at any time.

In this thesis, we will attempt to simplify the process of collecting data so that it can be done by anyone, anywhere, at any time. We will do this by training a counting network to recognize shells in an image and count them automatically. This is already done on a smaller scale by VLIZ with obsidentify. Obsidentify is a mobile app and website where users can submit pictures of a single shell and get a result of what kind of shell it is. This is a useful tool, but taking a close up picture of every single shell is a very time consuming process. We will have to work with a limited dataset to train the neural network as no dataset exist with large quantities of annotated pictures of beaches. After successful completion of this thesis, the new ideal scenario for collecting data can be found in figure 1.2. Compared to the current process, found in figure 1.1, this new process nearly eliminates the experts involvement and thus makes the process scalable.

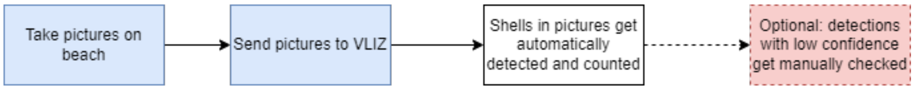


**Figure 1.2:** The new process of collecting data.

We will be studying if current counting networks are performant enough to recognize shells in beach image. We will build up to this by first training a network to count objects from a more established dataset in order to have a baseline to compare our model to. Afterwards we will then train that network with a small dataset to count shells and study its performance.

In the remainder of this thesis we will first discuss the state of the art in the field of object detection and counting, with a focus on few-shot learning. We will then discuss the datasets and the network architecture we will be using. In the second semester we will implement the network and train it on the datasets. We will then discuss the results and the limitations of our model. Finally we will discuss the future work that can be done to improve the model.

# Chapter 2

# Literature Review

In this chapter we will go over the state of the art in the field of object counting. We will go over the techniques commonly used for counting and go in more depth about the topic of few shot object detection and why it should be applied to our problem. Finally we will discuss the metrics used to evaluate the performance of the models.

## 2.1 (Crowd) Counting

Counting networks are quite an established concept in machine learning as numerous papers tackle the issue of counting humans, cars, animals or cells. What those have in common is that they only encompass a small set of possible categories to count and that, as they have a large real life use, large annotated datasets exist for these problems like ShanghaiTech and COWC. The problem we're trying to solve is a bit different as we want to count a large set of objects and we don't have a large dataset to train on.

The methodology behind counting networks has three big streams. The first one applies a detection method to the image and then counts the number of detected objects. Many different detection methods can be used, from looking for characteristic features to matching the shape of the objects. The second one takes a more global approach by first extracting features, textures, gradients and other information from the image as a whole and then using those to count the objects. The third method is not used on static images, but video. It assumes that the objects are moving in clusters and uses that to predict the movement of the objects and improve detection.

Out of those three methods, the second one is not applicable to our problem as the object we're trying to count are sparse and the third one is not applicable to our problem as we are trying to detect unmoving objects in a still image. The first method is thus the best option for our problem, but it still has a problem. Detection networks are trained on large datasets with a lot of images of the same object. This allows the model to learn the characteristics of the object and to detect it in a new image. As we don't have a large dataset, we can't use this method. We will have to use a different method to train our model, one that doesn't require a large dataset. This is where few shot object detection comes in. It allows us to train a model on a small dataset and still use it to detect objects in a new image.

## 2.2 Few-shot object detection

Few-shot object detection is a technique that has been gaining popularity in the last few years. It allow you to train a model on a small dataset, which is useful in scenarios where it isn't plausible to get a large dataset to train due to the cost or the time it would take to get it. The idea behind few-shot is as follows, if a human can recognize an object after seeing it a few times, then a machine should be able to do the same. A human achieves this because they have seen many types of objects and can use that knowledge to extract significant features of new object. In essence few-shot learning does the same. It relies on a large dataset of (similar) objects and then applies the knowledge gained from that to a new object, with fewer examples.

Practically this can be done in two different ways. The first way is taking a pre-trained model and transfer learning to finetune it on the new small dataset. The second way is meta learning, where the model learns to learn. This means that the model will learn how it should learn to recognize new objects from a large dataset to then apply that to the small dataset. Both of these methods have their characteristics, which we will go over in the next section.

### 2.2.1 Transfer learning

Transfer learning is a technique that has been used for a long time in machine learning. It allows you to use a model that has been trained on a large dataset as a base and finetune it on a new dataset. The training pipeline for this method is simple, as it works by freezing part of the base model and finetuning it on the new dataset.

### 2.2.2 Meta learning

Meta learning is a more recent technique in few-shot detection. The idea behind meta learning is that the model learns how to learn to detect new objects. For meta learning there are two options, single branch and dual branch meta learning. As dual branch meta learning is generally more effective, we will focus on that. Dual branch meta learning trains using two branches, a query branch and a support branch. The query branch contains full sized images, while the support set contains cropped images of a single labeled object. Both branches try to extract relevant features from their respective images with a shared backbone. The query branch then uses the features extracted from the support branch to detect the object in the query image. The loss is then propagated back through the backbone. This allows the model to learn how to learn to detect new objects.

## 2.3 Proposed approach

In this section we'll go into more detail about the approach we'll be taking. This will include a number of subsections, each describing a different aspect of the approach.

### 2.3.1 Few-shot

FACULTY OF ENGINEERING TECHNOLOGY
TECHNOLOGY CAMPUS DE NAYER
Jan De Nayerlaan 5
2860 SINT-KATELIJNE-WAVER, Belgium
tel. + 32 15 31 69 44
fet.denayer@kuleuven.be
www.fet.kuleuven.be