Lappeenranta University of Technology
School of Engineering Science
Master's programme in Computational Engineering and Technical Physics
Intelligent Computing Major

Master's Thesis

Anastasia Popova

CO-SEGMENTATION METHODS FOR WILDLIFE-PHOTO IDENTIFICATION

Examiners: Professor Heikki Kälviäinen
Docent, Dr. Yana Demyanenko

Supervisors: Professor Heikki Kälviäinen
Docent, Dr. Tuomas Eerola

ABSTRACT

Lappeenranta University of Technology
School of Engineering Science
Intelligent Computing Major

Anastasia Popova

Co-segmentation methods for wildlife-photo identification

Master's Thesis

2017

40 pages, 17 figures, 4 tables.

Examiners: Professor Heikki Kälviäinen
Docent, Dr. Yana Demyanenko

Keywords: segmentation, co-segmentation, databases, image processing, wildlife photo-ID, animal biometrics

Co-segmentation is defined as the task of jointly segmenting shared objects in a given set of images. This work is concentrated on the case when the segmented object is an animal, which means that the segmentation might require additional information about animal biometrics. The aim of this thesis was to survey the existing segmentation and co-segmentation methods with respect to the task of wildlife photo-identification, to overview existing datasets for co-segmentation and to evaluate and compare existing co-segmentation algorithms. In this study four co-segmentation algorithms were compared: Discriminative clustering, Multiple foreground co-segmentation, Multiple random walkers, and Distributed co-segmentation via submodular optimization. The comparison was performed using various datasets with wildlife animals. In most cases the Multiple random walkers method showed the best results.

CONTENTS

LIST OF ABBREVIATIONS

AWA	Animals with Attributes
CoSand	Distributed Co-segmentation via Submodular Optimization
DC	Discriminative Clustering
EM	Expectation maximization
GMM	Gaussian mixture model
LP	Linear Programming
MFC	Multiple Foreground Co-segmentation
MRF	Markov Random Field
MRW	Multiple Random Walkers
SLIC	Simple Linear Iterative Clustering
SPM	Spatial Pyramid Matching
SVM	Support Vector Machine

1 INTRODUCTION

1.1 Background

Wildlife photo-identification is a commonly used technique to identify and to track individuals of wild animal populations over time. It has various applications in behavior and population demography studies, including tracking the populations, migration patterns and general behavior of animal species. Nowadays, mostly due to large and labor-intensive image data sets, automated photo-identification is an emerging research topic.

A typical method to capture image material for the photo-identification is to use static camera traps. Therefore, the same animal individual is often captured with the same background. This increases the risk that a supervised identification algorithm learns to identify the background instead of the actual animal if the full image is used. To avoid this problem, it is useful to segment the animal from the background, using segmentation algorithms. Examples of the automatic segmentation of a seal are shown in Figure 1.

The basic idea in co-segmentation is to detect and to segment the common object (e.g., animal) in a set of images despite the different appearance of the object and different backgrounds. Such methods provide a promising approach to process large photo-identification databases for which manual or even semi-manual or supervised approaches are very time-consuming. However, it should be noted that automatic segmentation of animals is often difficult due to the camouflage colors of animals, i.e., the coloration and patterns are similar to the visual background of the animal.

In this thesis a review of existing co-segmentation methods and wild animal databases has been made. Four co-segmentation methods were selected for testing and evaluation: 1) Discriminative Clustering (DC) [1], 2) Multiple Foreground Co-segmentation (MFC) [2], Multiple Random Walkers (MRW) [3] and Distributed Co-segmentation via Submodular Optimization (CoSand) [4]. Experiments were performed using Saimaa Ringed Seals database [5] and the iCoseg database [2].

1.2 Objectives and restrictions

This Master's thesis has the following objectives:

- To survey existing co-segmentation methods with respect to the task of wildlife photo-identification.
- To overview existing databases which might be used for testing and evaluation of co-segmentation algorithms.
- To evaluate and to compare selected co-segmentation methods with various datasets.

In this study no other objects are considered except for wildlife animals. Multi-object approaches for co-segmentation were not used. For those co-segmentation methods that support multiple foreground objects single object case was used. No video co-segmentation approaches were tested. The photo-identification of animals was not considered.

1.3 Structure of the thesis

Rest of the thesis is organized as follows. Chapter 2 contains the formulation of a segmentation problem and overview of existing segmentation methods. Chapter 3 contains co-segmentation methods classification and their brief description. Chapter 4 contains the overview of existing co-segmentation datasets and wildlife animal datasets. Chapter 5 contains the experiments description and obtained results. Chapter 6 contains discussion on obtained results. Chapter 7 contains the conclusion and the directions of the future work.

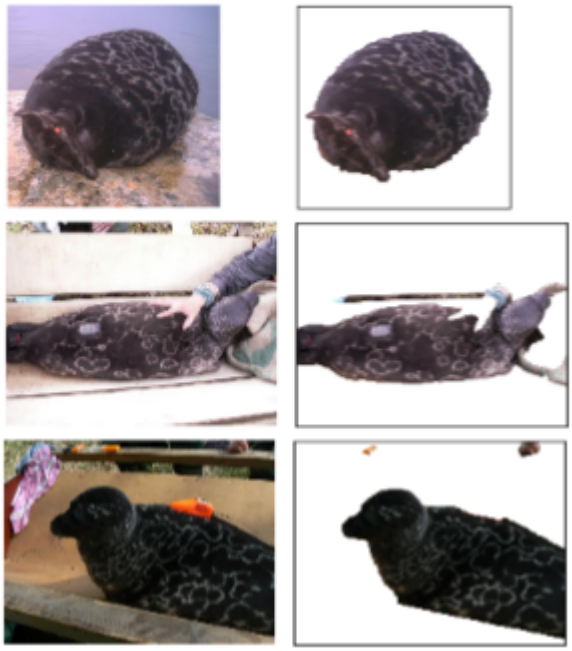


Figure 1. Example segmentation results (from the left to the right): the input image and the segmentation result. Modified from [5].

2 SEGMENTATION

Generally, segmentation in computer vision means partition of an image into constituent parts. Semantic segmentation means partitioning of the images into semantically meaningful segments. Although this operation is widely used in a large variety of applications (image annotation, content-based image retrieval, traffic control systems, etc.), automatic and precise segmentation problem is still unsolved. Existing approaches for solving the problem can be grouped into threshold-based, bottom-up methods, and interactive methods.

2.1 Threshold-based segmentation

Thresholding is the simplest method of image segmentation. Each pixel in the source image is assigned to two or more classes. This method of segmentation simultaneously applies a single fixed criterion to all pixels in the image. The criterion can be expressed as:

$$g(x,y) = \begin{cases} 0, \text{if } f(x,y) < T \\ 1, \text{if } f(x,y) \geq T \end{cases} \tag{1}$$

where (x, y) are the pixel coordinates, $f(x, y)$ is the gray-level value and T is the threshold. The selection of the threshold can be performed by a number of techniques. Based on the threshold selection approach thresholding algorithms are divided into three groups:

- Global thresholding where a single threshold value is used in the whole image.
- Local thresholding where the threshold value depends on gray-levels of $f(x, y)$ and local image properties of neighboring pixels.
- Adaptive thresholding where threshold is recalculated for each pixel in the image.

The example of threshold based segmentation is illustrated on Figure 2.

(a) (b)

Figure 2. An example of threshold based segmentation: a) Original image; b) Segmentation result. [6]

2.2 Bottom-up methods

Bottom-up methods are based on hierarchical grouping from low-level features to high level structures, using some local homogeneity of objects. Depending on how the image is interpreted, bottom-up methods might be separated into two categories: discrete and continuous methods [7].

Discrete methods treat an image as a fixed discrete grid. One of the simplest methods in this category is k-means [8]. This method segments an image into the predefined number of clusters, which are determined by k centroids. The centroids can be initialized by random points from the image. The algorithm contains two main steps: 1) assigning each point of image to the closest centroid and 2) recalculating each centroid as the mean of points assigned to the corresponding class. These two steps are repeated until the centroids stop changing or the limit of iterations is reached.

Another discrete bottom-up method is Mixture of Gaussians [8]. The idea of this method is similar to k-means. The centroids are replaced by a covariance matrix whose values are re-estimated from the corresponding samples. The clusters are represented as the mixture of Gaussians. Parameters of the mixture are re-estimated using expectation maximization (EM) algorithm. The association with the cluster center is realized using Mahalanobis distance.

Graph-based region merging [9] represents the image as a graph where each pixel denotes a node of the graph and neighboring pixels are connected by undirected edges. Weights are determined by the dissimilarity between pixels. Initially each node forms their own region. Then the regions are iteratively merged. Regions C_i and C_j are merged if in-

between edge weight is less than $\min(\int(C_i)+\tau(C_i), \int(C_j)+\tau(C_j))$, where $\tau(C) = \frac{k}{|C|}$ and k is a coefficient that is used to control the component size.

Continuous methods interpret an image as a continuous surface instead of a discrete grid. The example of such method is the snake model [10]. A snake is an energy-minimizing spline. The evolution of the snake is driven by minimizing the internal energy function:

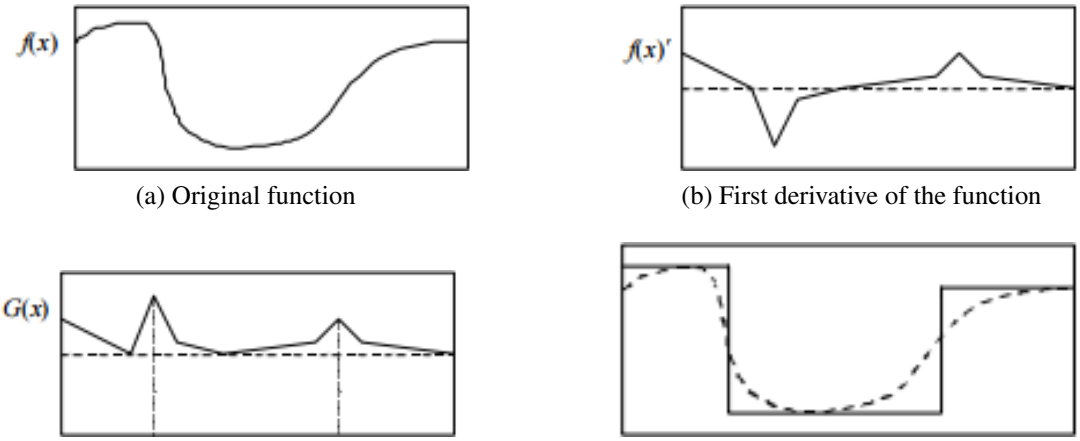
$$E^*_{snake} = \int_0^1 E_{snake}(v(s))ds = \int_0^1 E_{int}(v(s)) + E_{image}(v(s)) + E_{con}(v(s))ds, \qquad (2)$$

where E_{int} represent the internal energy of the spline, which is composed of the continuity of the contour and the smoothness of the contour E_{image} is the image constraint force, E_{con} is the constraint force introduced by the user.

Superpixel based methods aim to represent an image as a group of over-segmented pixel regions. These methods do not provide the final solution and are generally used for pre-processing() e.g. Toboggan-based methods [11]). The Toboggan method is designed to associate each pixel with the minimum of the valley where the pixel is located. First, the image is preprocessed to the gradient image. The first step is assigning each local-maximum flat region, according to the gradient information, with the unique label. Then all non-local-maximum flat regions are assigned with the label of the closest local-maximum flat region. Figure 3 illustrates the segmentation applied to a one-dimensional function $f(x)$. Figure 3(a) illustrates the original function and in Figure 3(b) is the first derivative. Figure 3(c) shows $G(x) = |f(x)'|$ and its local maxima which corresponds to separate classes. Figure 3(d) illustrates the segmentation result.

Normalized cuts method for image segmentation is based on selecting the partition on the image graph by minimization of goodness criterion [12]. The structure of the image graph G is similar to graph-based region merging approach, where each pixel corresponds to nodes of the graph (V) and neighboring pixels are connected by edges (V) and the weight on each edge $w(i, j)$) is the similarity function between the corresponding nodes i, j. A graph partition of a graph $G = (V; E)$ is defined as two disjoint sets A, B, $A \cup B = V$, $A \cap B = \emptyset$. Disassociation measure the normalized cut ($Ncut$)

$$Ncut(A, B) = \frac{cut(A, B)}{assoc(A, V)} + \frac{cut(A, B)}{assoc(B, V)}, \qquad (3)$$



(a) Original function

(b) First derivative of the function

(c) Local maximums of absolute values of the derivative

(d) Segmentation result

Figure 3. Toboggan method segmentation in one-dimensional case. Modified from [11].

where $cut(A, B) = \sum_{u \in A, v \in B} w(u, v)$, $assoc(BV) = \sum_{u \in A, t \in V} w(u, t)$. The segmentation algorithm represents finding the partition with minimal disassociation measure $Ncut(A, B)$. Generally finding minimizing normalized cut is NP-complete problem, but an approximate discrete solution can be found efficiently by formulating the minimization of criterion as a generalized eigenvalue problem. The eigenvectors can be used to construct good partitions of the image and the process can be continued recursively.

2.3 Interactive methods

Interactive methods require human intervention during the segmentation process. Generally, this means selecting the initial input and choosing either to stop after the segmentation iteration or to continue. Interactive methods might also be divided into contour based and label propagation methods. Contour based methods use edge detection algorithms for finding the closest contour to the area selected by a user (see e.g. Live-Wire [13, 14]). Label propagation technics use global optimization methods for propagating the initial area set by the user (GrabCut [15]). The examples of GrabCut segmentation are illustrated on Figure 4.

Figure 4. Examples of GrabCut. The user drags a rectangle loosely around an object. The object is then extracted automatically. Modified from [15].

3 CO-SEGMENTATION METHODS

The task of co-segmentation is to extract similar foreground objects from a set of images. The example of co-segmentation on seal images is illustrated on Figure 5. As the foreground object varies the main challenge is to define the similarity metric which goodness affects directly on the segmentation result. The existing approaches for co-segmentation can be generally divided into two categories [7]: 1) those that extend single-image segmentation methods with respect to the task of multiple images and 2) those that use models, for example based on clustering or graph theory.

3.1 Co-segmentation by extanding single-image segmentation

The general idea of extending single-image segmentation model can be formulated as minimizing the energy

$$E = E_s + E_g, \tag{4}$$

where E_s is the single image segmentation term which guarantees the smoothness and the distinction between foreground and background in each image, and E_g is the co-segmentation term which focuses on evaluating the consistency between the foregrounds among the images [7]. In number of classical approaches, the single image segmentation term is formed using Markov Random Field (MRF) on the graphs corresponding to the input image:

$$E_s^{MRF} = E_u^{MRF} + E_p^{MRF}, \tag{5}$$

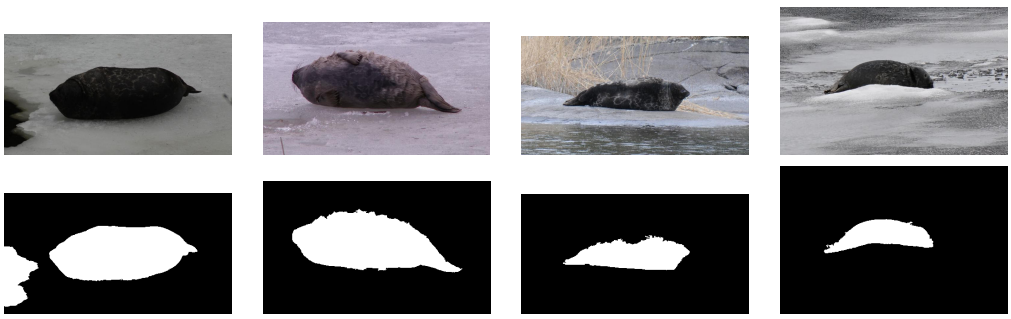


Figure 5. Example of co-segmentation on seal images.

where E_u^{MRF} and E_p^{MRF} are the conventional unary potential and the pairwise potential [16, 17]. the unary potential E_u^{MRF} is evaluated by using two Gaussian Mixture Models for background and foreground regions respectively. The pairwise potential E_p^{MRF} encourages coherence in regions of similar grey-level. For evaluating the consistency between the foregrounds among the images (co-segmentation term E_g) Rother et al. [16] used L_1 norm:

$$E_g = \sum_z (|h_1(z) - h_2(z)|) \tag{6}$$

where h_1 and h_2 are features of the two foregrounds, and z is the dimension of the feature. Mukherjee et al. [17] replaced L_1 with L_2 evaluation, i.e.

$$E_g = \sum_z (|h_1(z) - h_2(z)|)^2 \tag{7}$$

which leads to several advantages, such as relaxing the minimization to Linear Programming (LP) problem and using PseudoBoolean optimization [18] method for minimization. Another modification of the approach was introduced by Collins et al. [19, 3], where RandomWalk model was used instead of MRF segmentation model.

3.2 Model based co-segmentation

Model based co-segmentation use co-segmentation models rather than adapting single-image approaches. The task of co-segmentation is jointly partitioning same or similar object on a set of images.

Joulin et al. [1] proposed a method that is based on clustering strategy. Co-segmentation labeling is treated as training data for a supervised classifier. Then the classifier is trained with these labels until the maximal separation of the two classes is achieved. The co-segmentation is then formulated as searching of the labels that lead to the best classification.

Another clustering based approach was introduced in [20] where the images are divided into hierarchical superpixel layers where the relationships of the superpixels are described using graph. Then the affinity matrix is constructed and the co-segmentation problem is solved using spectral clustering [21].

Graph based approaches represent the region similarity relationships as edge weights of

a graph. In the method proposed by Vicente et al. [22] the graph is fully connected and the co-segmentation is achieved by loop belief propagation. Meng et al. [23] constructed directed graph structure to describe the foreground relationship by only considering the neighboring images. The object co-segmentation is then formulated as a shortest path problem [7].

3.3 Discriminative clustering for image co-segmentation

In [1] a discriminative clustering (DC) approach for co-segmentation was proposed, which is based on the combination of bottom-up image segmentation with kernel methods. DC is a technique for performing unsupervised clustering using support vector machine (SVM). The goal of the algorithm is to assign foreground and background labels jointly to all images, so that the SVM trained with these labels leads to maximal separation of the two classes.

For adapting DC to the task of co-segmentation in [1] spatial consistency constraint was introduced and an efficient convex relaxation approach was proposed for the hard-combinatorial optimization problem.

Suppose there is a set of images $I = I_1, ..., I_n$. W^i is the similarity matrix defined for each image where for any pair of pixels (l, m), W^i_{lm} is zero if the two pixels are separated by more than two nodes in the image grid, otherwise W^i_{lm} is given by:

$$W^i_{lm} = \exp(-\lambda_p ||p^m - p^l||^2 - \lambda_c ||c^m - c^l||^2). \tag{8}$$

The Laplacian matrix L is defined as follows:

$$L = I_n - D^{-1/2} W D^{-1/2}, \tag{9}$$

where W is constructed by assembling the separate similarity matrices $W^i, i = 1, ..., q$ into a block-diagonal matrix $W \in R_n \times n$, by putting the blocks $Wi \in R_n \times n$ on the diagonal, D is the diagonal matrix composed of the row sums of W.

Given the normalized Laplacian matrix, a spectral method like normalized cuts [22] outputs the second smallest eigenvector of L Normalized cuts [12] method is used for obtaining the the second smallest eigenvector of L and the obtained normalized cuts are included

as the term the objective function. A joint $n \times n$ positive semidefinite kernel matrix K is defined is follows:

$$K_{lm} = \exp(-\lambda_h \sum_{d=1}^{k} \frac{(x_d^l - x_d^m)^2}{x_d^l - x_d^m}), \tag{10}$$

where x is a k-dimensional feature vector (features are color histograms and Gabor features), $\lambda_h > 0$. In [1] $\lambda_h = 0.1$ was used.

Supervised classifier is an affine function of feature map which is applied for learning through the minimization with respect to $f \in F$, where F is a high-dimensional Hilbert space and $b \in \mathbb{R}$ of

$$\frac{1}{n} \sum_{j=1}^{n} l(y_i, f^T \Phi(x^j) + b) + \lambda_k ||f||^2, \tag{11}$$

where $y_j \in \{-1, 1\}$ is the label associated with the j-th pixel and l is the loss function $l(s,t) = (l-t)^2$ [24]. The measure of the separability of the classes defined by $y_j \in \{-1, 1\}^n$ is the optimal solution of the supervised learning problem in (11). The main steps of the algorithm are summarized in Algorithm 1.

Algorithm 1: Discriminative clustering (DC) [3].

Input : Set of input images I
Output: Segmentation maps $C = \{C_1, ..., C_Z\}$

```
foreach image I_i do
    Compute similarity matrix W_i;
end
Construct W by assembling similarity matrices W_i;
Compute Laplacian matrix L using normalized cuts;
Construct semidefinite kernel matrix K;
Learn a classifier through the minimization using kernel methods;
foreach image I_i do
    foreach pixel (l, m) do
        c_lm is obtained through classification using the trained classifier;
    end
end
```

3.4 Multiple Foreground Co-segmentation

In [2], a method for multiple foreground co-segmentation (MFC) was proposed. The task of MFC is defined as the task of joint segmentation of K different foregrounds $F = F_1, ..., F_K$ from M input images, each of which contains a different unknown subset of K foregrounds. The proposed approach supports two different scenarios: 1) unsupervised scenario where a user specifies only the number of foregrounds K and the algorithm automatically distinguish K foregrounds that are most dominant in I 2) supervised scenario where a user provides bounding-box or pixel-wise annotations for K foregrounds of interest in some selected images. [2] In this study only unsupervised scenario was considered.

The proposed MFC approach consists of two modules: 1) foreground modeling module; 2) region assignment module. A parametric function $v^k : S \to \mathbb{R}$ represents the model of the foreground F_k. v^k maps any region $S \in \boldsymbol{S}$ in an image to its fitness value to the k-th foreground. If S_i is the oversegmented representation of I_i then $v^k : 2^{|S_i|} \to \mathbb{R}$ takes any subset $S \in \boldsymbol{S}_i$ as an input and returns its value to the k-th foreground. In this approach function v^k is defined by the Gaussian mixture model (GMM) (i.e. BoykovJolly model [25, 15]) and spatial pyramid matching (SPM) with linear support vector machine (SVM) [26].

First step of the algorithm is oversegmentation of each image by applying submodular image segmentation [4]. Suppose I_i is the image and S_i is the set of its oversegmented regions. Then each region from S_i is assigned separately using the given foreground model, which provides disjoint subsets of foregrounds F_i^k ($k = 1, ..., K$) and background. Finally, the task reduces to finding a disjoint partition $S_i = \cup_{k=1}^{K+1} F_i^k$ with $F_i^k \cap F_i^l = \emptyset$ if $k \neq l$, to maximize the following sum:

$$\sum_{k=1}^{K+1} v^k(F_i^k) \tag{12}$$

The detailed scheme of the method is presented in Algorithm 2.

3.5 Distributed Co-segmentation via Submodular Optimization

In [4] a distributed co-segmentation approach (CoSand) for was presented. The main feature of this approach is the ability to cope with a highly variable large-scale image

Algorithm 2: Multiple foreground co-segmentation [2]

Input : (1) Input image set I. (2) Number of foregrounds (FGs) K.
Output: Foregrounds $F = F_1, ..., F_K$ for all $I_i \in I$

Initialization: **foreach** $I_i \in I$ **do**
  Oversegment I_i to S_i and build adjacency graph $G_i = (S_i, E_i)$ where $(s_l, s_m) \in E_i$ if $\min d(s_l, s_m) \leq \rho$. Apply diversity ranking to the similarity graph of $S = \cup_{i=1}^{M} S_i$ to find K regions $A = \{A_1, ..., A_K\}$ that are highly repeated in S and diverse with respect to each other.
end
Set $F \leftarrow A$ Iterative Optimization: The iteration stops if a new region assignment does not increase the objective value $\sum_{i=1}^{M} \sum_{k=1}^{K+1} v^k(F_i^k)$ Foreground Modeling: **foreach** $k \in 1 : K$ **do**
  Learn GMM and SPM FG models from F^k
end
Region assignment: **foreach** $I_i \in I$ **do**
  foreach $k \in 1 : K + 1$ **do**
    Generate FG candidates B_i^k as a set of $B_i^k = \{k_j, C_j, w_j\}$, where k_j is the foreground index, $C_j \in S_i$ is a subtree of G_i and $w_j = v^k(C_j)$
  end
  Compute the most probable candidate tree T_i^* and pruned B_i^* from $B_i = \cup_{k=1}^{K+} B_i^k$
  Obtain F_i to solve region assignment by using dynamic programming on B^*
end

collection.

The segmentation task is modeled by temperature maximization on anisotropic heat diffusion. The temperature maximization with K heat sources corresponds to a K-way segmentation that maximizes the segmentation confidence of every pixel in an image.

Segmentation of a single image is divided into several steps. First, superpixels are extracted, using TurboPixels [27]. Then the intra-image graph $G_i = (V_i, E_i, D_i)$ is constructed, where the vertex set V_i is the set of superpixels and the edge set E_i connects all pairs of adjacent superpixels. In each superpixel, 3-D CIE Lab color and 4-D texture features are extracted. [4] D_i is the diffusivity which is computed by Gaussian similarity on the features of superpixels:

$$d_{xy} = \begin{cases} -exp(\beta||g(x) - g(y)||^2), \text{ if } x, y \in E_i \\ 0, \text{ otherwise} \end{cases}, \tag{13}$$

where $g(x)$ is a feature vector in a node $x \in V_i$. Final step is agglomerative clustering [4] on G_i to find out the set of evaluation points L_i, where the algorithm greedily selects the

largest and most coherent regions. As the iteration goes, the previous results are re-used, which significantly reduce the computation time (e.g. the lazy greedy approach in [28]).

The optimization formulation for co-segmentation is an extension of the diversity ranking [29] on G_i, where the objective is the sum of segmentation confidence of every image in the dataset. This formulation encourages each image to be segmented as K regions which are the most coherent, content-wise diverse with respect to one another and the largest. The main structure of the algorithm is summarized in Algorithm 3.

Algorithm 3: Distributed Co-segmentation via Submodular Optimization (CoSand) [4]

Input : (1) Intra-image matrix G_i for all $I_i \in I$. (2) Number of segments K. (3) Evaluation set size $|L|$.

Output: Cluster centers S_i and segmented images for $I_i \in I$

foreach $I_i \in I$ **do**
  $S_i \leftarrow \emptyset$
end
foreach $I_i \in I$ **do**
  $I_i \in I$ do $L_i \leftarrow$ agglomerative clustering on G_i
end
while $|S_i| \leq K$ **do**
  foreach $I_i \in I$ **do**
    foreach $l_j \in L_i$ **do**
      Solve $u = L_i u$ where L_i is the Laplacian of G_i and u is an $N_i \times 1$ vector with the constraints of $u(S_i \cup l_j) = 1$ and $u(g) = 0$. Obtain the gain $\triangle U_i(l_j) = |u|_1$ (l-1 norm of u)
    end
  end
  Solve the energy maximization by belief propagation $E(l) = \sum_{i \in I} \triangle U_i(l_i)(\frac{1}{N(i)} \sum_{j \in N(i)} f(g(l_i), g(l_j)))$, $s_1, ..., s_I \leftarrow argmax_{1,...l_I} E(l)$, where f is the Gaussian similarity.
  foreach $I_i \in I$ **do**
    $S_i \leftarrow S_i \cup s_i$
  end
end
foreach $I_i \in I$ **do**
  Compute $(N_i - K) \times K$ matrix X by solving $L_u X = -B^T I_s$ where $X_i = \frac{V_i}{,} S_i, L_u = L_i(X_i, X_i)$, $B = L_i(S_i, X_i)$, and I_s is a $K \times K$ identify matrix. A superpixel $v_j (\in V_i)$ is clustered c_j = $argmax_k X(j, k)$.
end

3.6 Co-segmentation using multiple random walkers

Another approach for image co-segmentation is a graph-based system to simulate the movements and interactions of multiple random walkers (MRW) [3].

Generally, in [3] a random walk is described as a process in which a walker moves randomly from one node to another in a graph. The conventional random walk is a walk of a single random walker (or agent), which is described as a Markov process. Let $G = (V, E)$ be a weighted undirected graph. V is the set of nodes which corresponds to the data points x_i, $i = 1, ..., N$. Edge $e_{ij} \in E$ connects x_i and x_j. Let $W \in \mathbb{R}^{NxN}$ be a symmetric matrix, in which the (i, j)th element w_{ij} is the weight of e_{ij}, corresponding to the affinity between x_i and x_j. The transition probability a_{ij} of a random walker is the probability that the walker moves from node j to node i on the graph G. a_{ij} is obtained by dividing w_{ij} by the degree of node j, i.e., $a_{ij} = w_{ij} / \sum_k w_{kj}$. The temporal recursion of the transaction probabilities is described as:

$$p^{(t+1)} = Ap^{(t)}, \tag{14}$$

where $p^{(t)} = [p_1^{(t)}, ...p_N^{(t)}]^T$ denotes the probability that the walker is found at node i at time instance t, $A = [a_{ij}]$ is the transition matrix, computed by normalizing each column of the matrix W. If the graph G is fully connected and has a finite number of nodes, A is irreducible and primitive [30]. Then, regardless of an initial condition $p^{(0)}$ the walkers has a unique stationary distribution Π satisfying $\Pi = A\Pi$ and $\Pi = lim_{t\to\infty} = p^{(t)}$. The stationary distribution Π conveys useful information about the underlying data structure of the graph [31].

Suppose there are K agents on a graph, then $p_k^{(t)}$ is the probability distribution of agent k at time t. Similar to (14), random movements of agent k are defined as follows:

$$p_k^{(t+1)} = (1 - \epsilon)Ap_k^{(t)} + \epsilon r_k^{(t)}, k = 1, ..., K.[3] \tag{15}$$

The interaction of random walkers is determined by determining the restart distribution as

$$r_k^{(t)} = (1 - \delta^t)r_k^{(t-1)} + \delta^t \phi_k(\rho^{(t)})[3], \tag{16}$$

where the function ϕ_k is referred to as the restart rule. It determines a probability distri-

bution ϕ_k from $\rho^{(t)} = \{p^(t)_k\}_{k=1}^K$, which is the set of the probability distributions of all agents at time t.

By designing the restart rule ϕ_k in (16), a variety of agent interactions might be simulated to achieve a desired goal. In [3] the repulsive restart rule was used for clustering data.

To cluster images using random walkers, the graph G is constructed with nodes V corresponding to superpixels, which are obtained using simple linear iterative clustering (SLIC) superpixels [32]. For the edge set E, the edge connection scheme from [33] is used. For each edge e_{ij}, the affinity weight w_{ij} is defined by employing the dissimilarity function

$$d(x_i, x_j) = \sum_l \lambda_l d_l(x_i, x_j), \tag{17}$$

where $x_i, x_j \in V$ and e_{ij} connects x_i and x_j, λ is empirically determined weight [3]. Five dissimilarities d_l of node features are proposed in [33], including RGB and LAB superpixel means, boundary cues, bag-ofvisual-words histograms of RGB and LAB colors [34]. The main steps of the approach are presented in Algorithm 4.

Algorithm 4: Multiple Random Walkers (MRW) [3].

Input : Graphs $G = \{G_1, ..., G_Z\}$ for a set of input images I
Output: Segmentation maps $C = \{C_1, ..., C_Z\}$

```
Initialize P_(u) = {p_f(u), p_b(u)} for each I_u
repeat
    foreach image I_u do
        Compute inter-image concurrence;
        Cluster intra-image MRW;
        Extract foreground C = C_1, ..., C_Z;
        Compute the foreground distance Σ_{u,v} d_f(C_u, C_v) ;
    end
until The foreground distance stops decreasing;
Refine pixel-level.
```

4 EXISTING DATABASES

4.1 Co-segmentation databases

The iCoseg dataset [35] is a large binary-class image co-segmentation dataset, which contains 38 groups with a total of 643 images. The content of the images includes a wide range of objects including five groups of wild animals (bears, elephants, geese, cheetahs, pandas). Each group contains images of similar object instances as well as instances where objects are deformed considerably in terms of viewpoint and illumination, and in some cases, only a part of the object is visible. The examples of images of the database are illustrated in Figure 6.

The FlickrMFC dataset [2] is the only dataset for multiple foreground co-segmentation, which consists of 14 groups of manually labeled image. The image content covers daily scenarios such as children-playing, fishing and sports. The content also includes a number of image sets with animals (gorillas, cows, dogs, dolphins, parrots, swans). The examples of animal images from the FlickrMFC dataset are illustrated in Figure 7. This dataset contains a number of repeating subjects that are not necessarily presented in every image. Some images include strong occlusions, lighting variations, or scale or pose changes.

The MSRC co-segmentation dataset [16] has been used to evaluate image pair binary co-segmentation. The dataset contains 25 image pairs with similar foreground objects but heterogeneous backgrounds. Some pairs of the images are picked such that they contain some camouflage to balance database bias which forms the baseline co-segmentation dataset.

4.2 Manually segmented wild animal databases

The Saimaa ringed seals database is a unique photo-ID database of Saimaa ringed seal images collected by University of Eastern Finland for the SealVision project. A subset of the images has been manually segmented [5]. The database contains 1044 images of Saimaa ringed seals and the total amount of manually segmented images is 392 (Figure 8). Most of the images contain one individual Saimaa ringed seal, and only few images contain two or more individuals [5].

Figure 6. Examples of images from the iCoseg dataset. [36]

Figure 7. Examples of images from FlickrMFC dataset. [37]

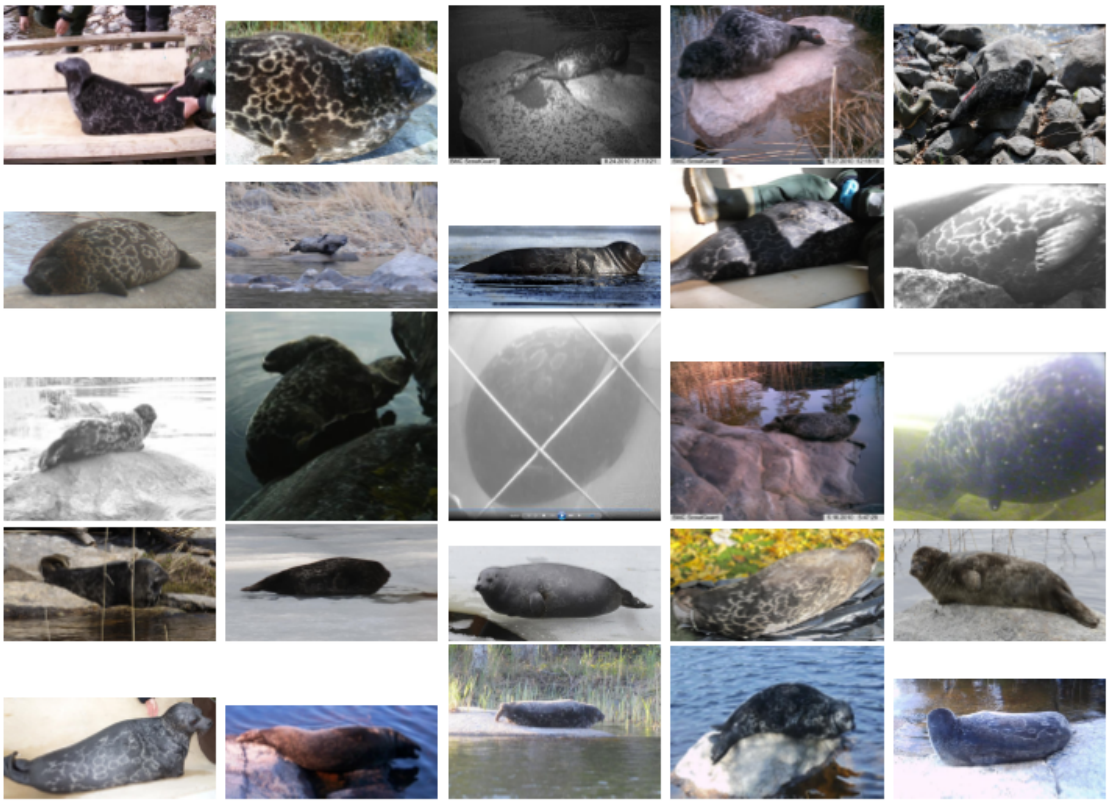


Figure 8. Examples of images from the Saimaa ringed seals database. [5]

4.3 Wild animal databases

Animals with Attributes (AWA) dataset [38] is a wild animal image dataset, which contains over 30,000 animals images of 50 animal types with six pre-extracted feature representations for each image. Examples of animal images are displayed in Figure 9.

The wild zebra database from field photographs [39] is a large zebra database used for creation of a biometric database of individual zebras differentiated by their coat markings. The database includes zebra photographs of two different population and the metadata of photographs with the same zebra individual, generated by a pattern recognition algorithm.

Neither AWA nor wild zebra database does not contain manual segmentation of animals, so they are not applicable for evaluation of co-segmentation methods, but they still can be used for visualization of results and manual analysis of the methods.

Figure 9. Examples of images from the AWA dataset, modified from [38].

5 EXPERIMENTS

5.1 Datasets

The comparison of the selected co-segmentation methods was performed using three subsets of the manually segmented Saimaa Ringed Seals dataset which were selected in increasing complexity. The first subset, called the SealVision easy subset includes five relatively simple images for co-segmentation with clear contrast between seal and background. The illumination is similar and there are no large obstacles in the images such as hands, tree branches or stones. The second subset, called the SealVision medium dataset was designed to be more realistic. It includes lighting variations, obstacles like hands and tree branches, but the images still have clear contrast between seal and background. The last subset, called the SealVision hard dataset contains all types of seals snapshots, including the most challenging ones with strong obstacles and grayscale night time images where the boundary of an animal is almost invisible.

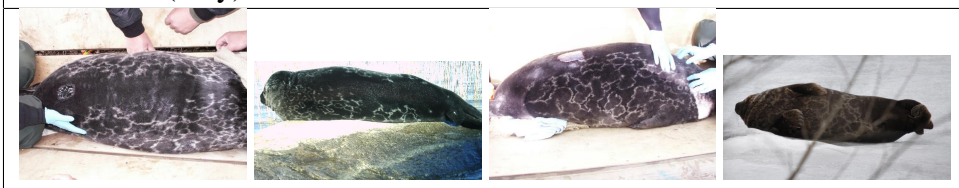
Another group of experiments was performed using animal images from the iCoseg dataset. These groups contain strong obstacles, but the illumination variation is relatively small. The characteristics of the datasets are summarized in Table 1. Table 2 illustrates the example images from each dataset.

Table 1. Characteristics of the datasets used in the experiments.

	SealVision			iCoseg		
	Easy	Medium	Hard	Alaskan brown bear	Elephants	Goose
Number of images	5	20	239	20	16	32
Multiple animals in one image	-	-	-	+	+	+
Strong obstacles	-	-	+	-	-	-
Minor obstacles	-	+	+	+	+	+
Night time images	-	-	+	-	-	-
Illumination variation	-	+	+	+	+	-

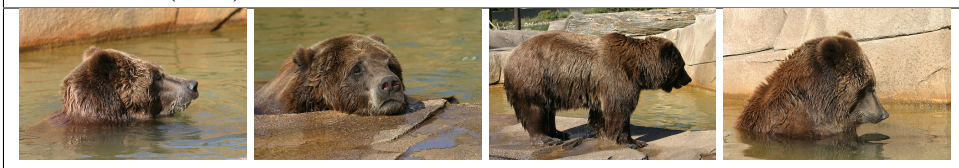
Table 2. Examples of images from the datasets used in experiments.

SealVision (easy)



SealVision (medium)

SealVision (hard)



iCoseg (alaskan brown bear)

iCoseg (goose)

iCoseg (elephants)

5.2 Evaluation criteria

For evaluation of co-segmentation methods each obtained binary matrix was checked for the similarity with the corresponding ground truth. For measuring similarity the Jaccard measure [40] was chosen which is a common measure for segmentation performance. Suppose that the image I has ground truth I_{GT}, represented as binary mask and I_S is the segmentation result of image I which is needed to be evaluated. Then the definition of the Jaccard measure is

$$S_{Jaccard} = \frac{|I_S \cap I_{GT}|}{|I_S \cup I_{GT}|}. \tag{18}$$

Figure 10 illustrates $I_S \cap I_{GT}$ and $I_S \cup I_{GT}$ in segmented seal picture compared with the ground truth. The Jaccard measure for this case is 0.74.

(a) (b) (c)

Figure 10. Visualization of Jaccard measure calculation: a) Ground truth (I_{GT}); b) Segmentation example (I_S); c) White area is $I_S \cap I_{GT}$, union of white and gray area represent $I_S \cup I_{GT}$.

The correct segmentation is defined based on the threshold of the Jaccard measure. For example, if the threshold is 0.6 then at least 60% of a segmented image must match the ground truth. To evaluate the performance of a co-segmentation method the number of correctly segmented images ($N_{correct}$) was calculated with respect to different thresholds of the Jaccard measure. As the threshold increases $N_{correct}$ decrease, so the better the method, the less the number $N_{correct}$ decreases with the increasing threshold.

5.3 Description of experiments

Four co-segmentation methods were chosen for the accuracy comparison: 1) Discriminative Clustering (DC) [1], 2) Multiple Foreground Co-segmentation (MFC) [2], Multiple

Random Walkers (MRW) [3] and Distributed Co-segmentation via Submodular Optimization (CoSand) [4], described in Chapter 3. To measure the similarity with the ground truth, presented as manually segmented binary masks, the Jaccard measure (18) was used. As the evaluation by this criterion requires the binary mask of the segmented images, co-segmentation results were processed for achieving the binary matrices.

The DC approach provides results as a matrix with three types of labels: foreground, background, and unknown pixels. For obtaining the binary format all the unknown pixels were set as the background. The selected number of classes was two, as the majority of test images contain only one animal, which means separating one foreground class from the background.

CoSand requires the predefined number of clusters k. Experiments showed that in the testing dataset $k = 3$ perform better than $k = 2$, as the algorithm tends to segment the brightest object in an image, such as blue water. The output of the algorithm represents a labeled image with k number of labels. Typically, an animal is presented in the center of images so for obtaining the binary format, the cluster which intersects the borders of the image less was selected as the foreground.

The MFC and MRW approaches provide directly the binary mask so the processing of the resulting images was not needed.

5.4 Results

Each selected co-segmentation method was tested using datasets described in Section 4.1. Examples of obtained co-segmentation results are shown in Table 3. Table 4 shows the comparison of the mean values of the Jaccard measure for obtained results for each dataset. Figures 11-16 show the percentage of images which were segmented correctly with respect to increasing threshold for each dataset and Figure 17 shows the overall result for six subsets combined together. Figures 11-16 show that in all six cases with different datasets the MRW algorithm achieved the best results. The mean Jaccard measure is the highest for each dataset (see Table 4). DC and MFC show relatively similar results, except for the SealVision medium dataset (Figure 12) and the iCoseg bears subset (Figure 16) where DC performs better than MFC. CoSand performs better than DC and MFC on the SealVision easy dataset (Figure 11), but in other cases CoSand performed relatively similar to MFC and DC.

Table 3. Examples of co-segmentation using selected methods.

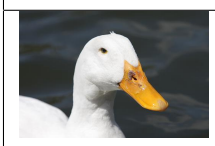
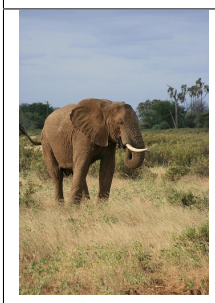
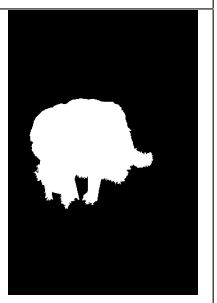
original image	DC	MFC	CoSand	MRW
				
SealVision (easy)				
				
SealVision (medium)				
				
SealVision (hard)				
				
iCoseg (alaskan brown bear)				
				
iCoseg (goose)				
				
iCoseg (elephants)				

Table 4. The comparison of the mean Jaccard measure of co-segmentation results compared with the ground truth.

	MRW	DC	MFC	CoSand
SealVision (easy)	**0.89**	0.59	0.48	0.78
SealVision (medium)	**0.87**	0.66	0.39	0.41
SealVision (hard)	**0.48**	0.21	0.18	0.17
iCoseg (alaskan brown bear)	**0.56**	0.47	0.13	0.18
iCoseg (elephants)	**0.67**	0.31	0.25	0.21
iCoseg (goose)	**0.69**	0.39	0.36	0.24
All the datasets combined	**0.54**	0.29	0.21	0.20

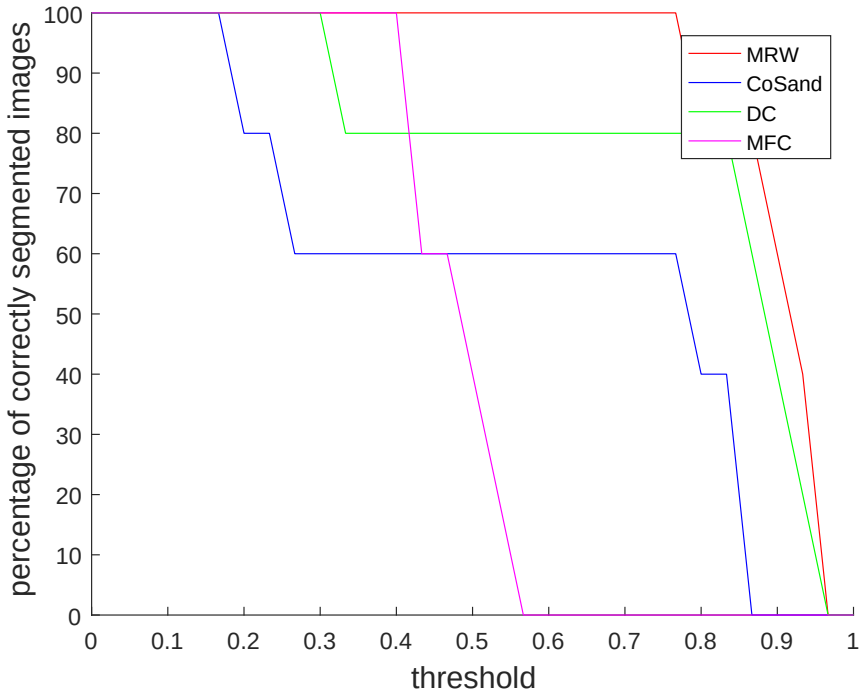


Figure 11. Percentage of correctly segmented images with respect to increasing threshold on the Saimaa ringed seals easy subset.

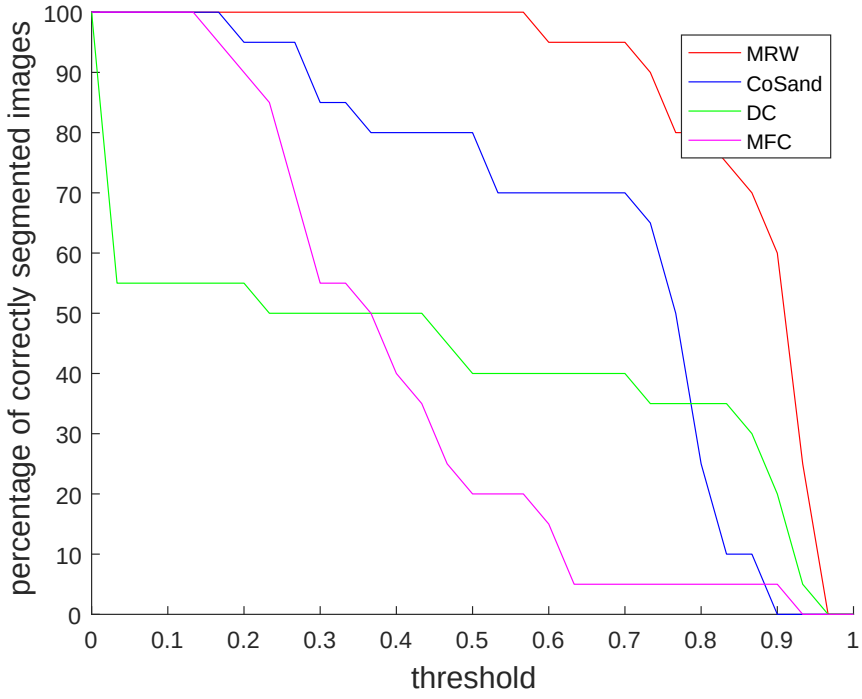


Figure 12. Percentage of correctly segmented images with respect to increasing threshold on the Saimaa ringed seals medium subset.

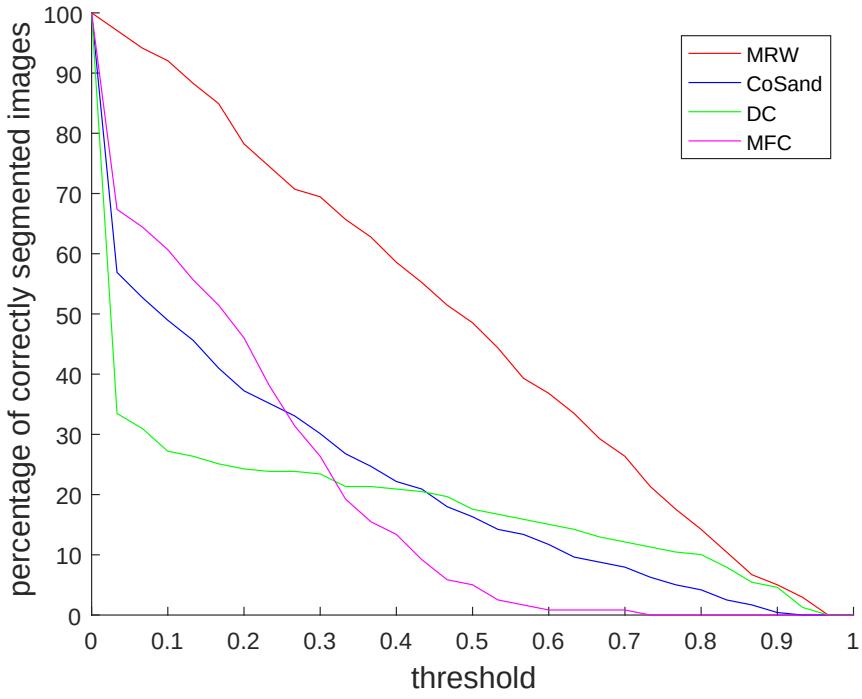


Figure 13. Percentage of correctly segmented images with respect to increasing threshold on the Saimaa ringed seals hard subset.

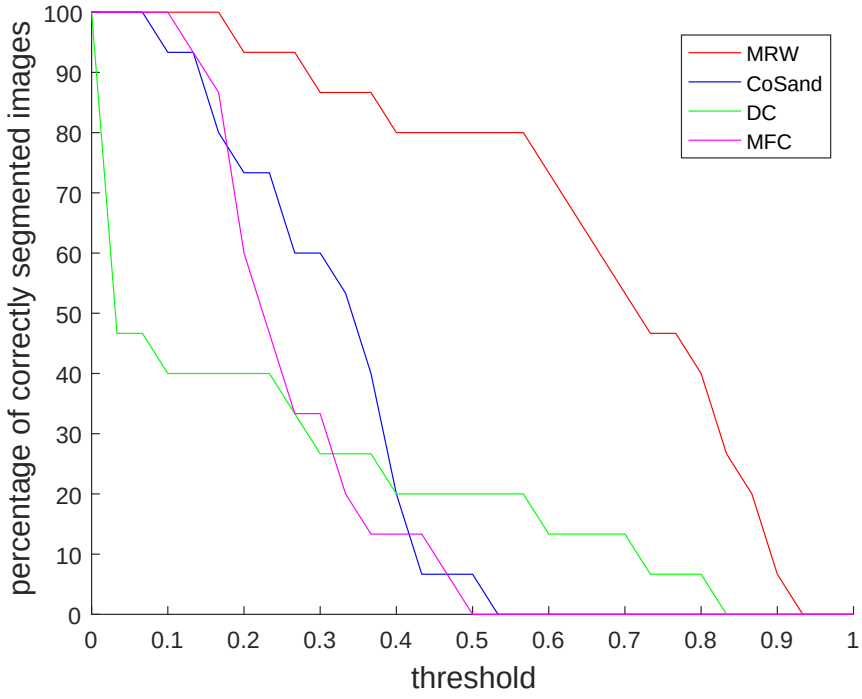


Figure 14. Percentage of correctly segmented images with respect to increasing threshold on the iCoseg subset with elephants.

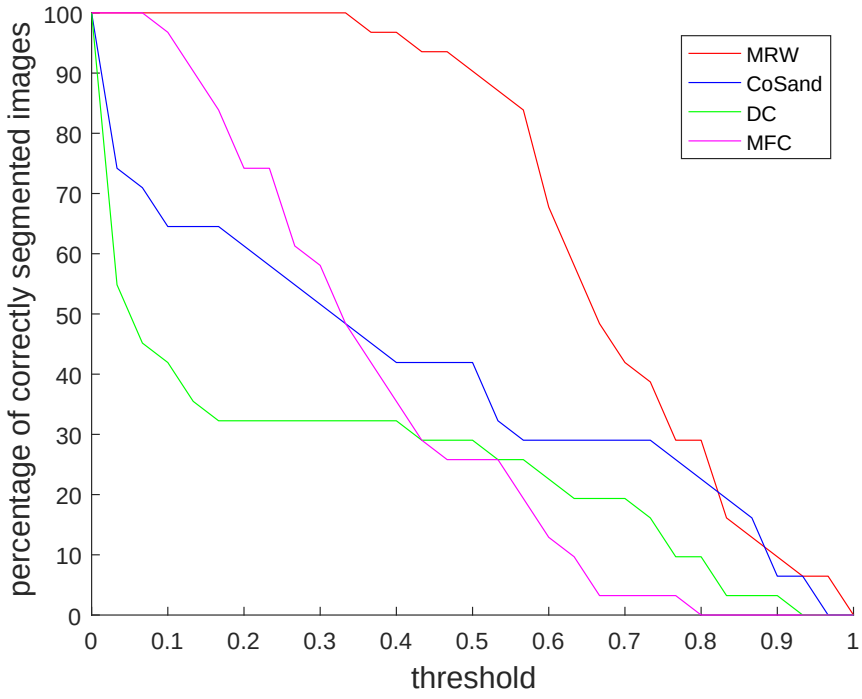


Figure 15. Percentage of correctly segmented images with respect to increasing threshold on the iCoseg subset with goose.

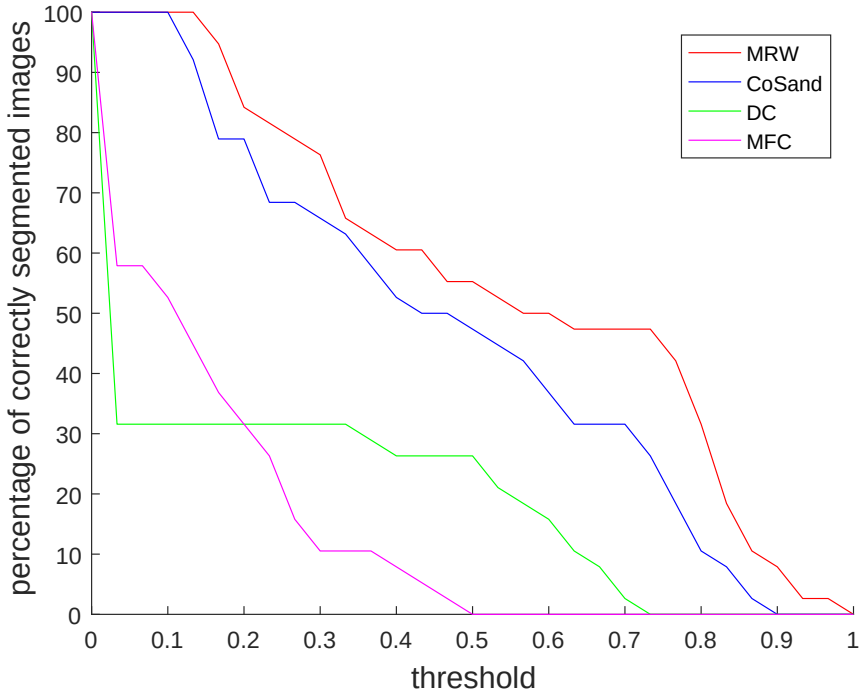


Figure 16. Percentage of correctly segmented images with respect to increasing threshold on the iCoseg subset with bears.

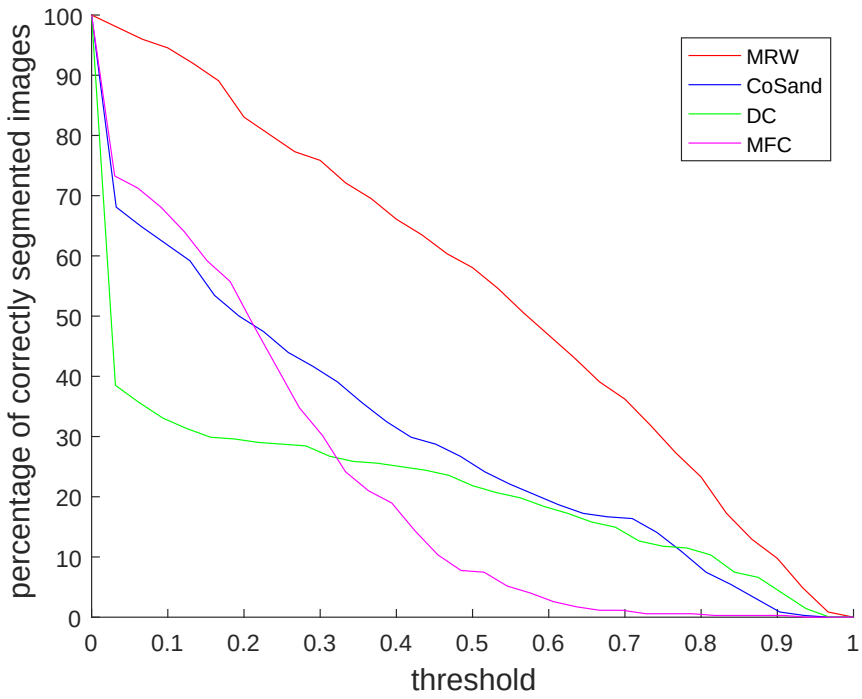


Figure 17. Percentage of correctly segmented images with respect to increasing threshold on all six subsets combined together.

6 DISCUSSION

In this study four co-segmentation methods were tested: 1) Discriminative Clustering (DC) [1], 2) Multiple Foreground Co-segmentation (MFC) [2], Multiple Random Walkers (MRW) [3] and Distributed Co-segmentation via Submodular Optimization (CoSand) [4]. Testing was performed on wild animal datasets described in Section 5. Jaccard measure was used for similarity measure between the segmentation result and the ground truth. The experiments showed that the Multiple random walkers approach performed better then other three algorithms in all six cases of different datasets (see Figures 11-17). The mean Jaccard measure is the highest in each case (see Table 4) and exceeds 0.5 for each subset.

DC showed high mean Jaccard measures on the SealVision easy and the medium subsets (0.5911 and 0.6650 respectively), but in other cases the mean Jaccard measure was lower than 0.5. According to percentage of correctly segmented images, DC performed better than CoSand and MFC on the SealVision medium subset (Figure 12) and the iCoseg bears subset (Figure 16). In other cases the accuracy was relatively similar to CoSand and MFC.

CoSand achieved higher mean Jaccard measure than DC and MFC on the SealVision easy subset (0.7784), but in all other cases the Jaccard measure was less than 0.5. According to percentage of correctly segmented images, CoSand performed better than DC and MFC on the SealVision easy subset (Figure 11), but in other cases the accuracy was close to MFC.

As it can be seen from Table 4 and Figures 11-17 MFC performed relatively similar to DC and CoSand, but the mean Jaccard measure was less than 0.5 on each dataset.

In the future work co-segmentation methods can be further analyzed by considering multiple objects in image. The performance can be improved by enhancing input images, for example by increasing contrast. Another direction of development is applying co-segmentation approaches on video for tracking purposes.

7 CONCLUSION

The objective of this study was to survey existing co-segmentation methods with respect to the task of wildlife photo-identification, to overview existing databases for the testing and evaluating the methods and to compare publicly available co-segmentation methods.

In the experiments two databases with manually segmented ground truth were used: 1) iCoseg, 2) Saimaa Ringed Seals database (SealVision). For detailed analysis of methods three subset were selected in increasing complexity: 1) SealVision easy subset, 2) SealVision medium subset, 3) SealVision hard subset.

Four methods were selected for the experiments: 1) Discriminative Clustering (DC) [1], 2) Multiple Foreground Co-segmentation (MFC) [2], Multiple Random Walkers (MRW) [3] and Distributed Co-segmentation via Submodular Optimization (CoSand) [4]. The Jaccard measure was used for calculation of similarity between the ground truths and the output binary masks. The results showed that in all cases MRW performed better than other three methods.

REFERENCES

[1] A. Joulin, F. Bach, and J. Ponce. Discriminative clustering for image co-segmentation. In *Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010.

[2] Gunhee Kim and Eric P. Xing. On Multiple Foreground Cosegmentation. In *Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[3] Chulwoo Lee, Won-Dong Jang, Jae-Young Sim, and Chang-Su Kim. Multiple random walkers and their application to image cosegmentation. In *Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3837–3845, 2015.

[4] Gunhee Kim, Eric P. Xing, Li Fei-Fei, and Takeo Kanade. Distributed Cosegmentation via Submodular Optimization on Anisotropic Diffusion. In *International Conference on Computer Vision (ICCV)*, 2011.

[5] A. Zhelezniakov, T. Eerola, M. Koivuniemi, M. Auttila, R. Levänen, M. Niemi, M. Kunnasranta, and H. Kälviäinen. Segmentation of saimaa ringed seals for identification purposes. In *Advances in Visual Computing, Springer Lecture Notes in Computer Science, LNCS*, volume 9475, pages 227–236. Las Vegas, USA, 2015.

[6] Thresholding (image processing). `.http://en.wikipedia.org/w/index.php?title=Thresholding_(image_processing)&oldid=606970852`. Accessed: 2017-05-21.

[7] Hongyuan Zhu, Fanman Meng, Jianfei Cai, and Shijian Lu. Beyond pixels: A comprehensive survey from bottom-up to semantic image segmentation and cosegmentation. *Journal of Visual Communication and Image Representation*, 34:12–27, 2016.

[8] Richard Szeliski. *Computer vision: algorithms and applications*. Springer Science & Business Media, 2010.

[9] Pedro F Felzenszwalb and Daniel P Huttenlocher. Efficient graph-based image segmentation. *International Journal of Computer Vision*, 59(2):167–181, 2004.

[10] Michael Kass, Andrew Witkin, and Demetri Terzopoulos. Snakes: Active contour models. *International Journal of Computer Vision*, 1(4):321–331, 1988.

[11] Eric N Mortensen and William A Barrett. Toboggan-based intelligent scissors with a four-parameter edge model. In *Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 2, pages 452–458. IEEE, 1999.

[12] Jianbo Shi and Jitendra Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, 2000.

[13] Eric Mortensen, Bryan Morse, William Barrett, and Jayaram Udupa. Adaptive boundary detection using'live-wire'two-dimensional dynamic programming. In *Proceedings of Computers in Cardiology*. IEEE, 1992.

[14] Eric N Mortensen and William A Barrett. Intelligent scissors for image composition. In *Proceedings of the 22nd Annual Conference on Computer Graphics and Interactive Techniques*, pages 191–198. ACM, 1995.

[15] Carsten Rother, Vladimir Kolmogorov, and Andrew Blake. Grabcut: Interactive foreground extraction using iterated graph cuts. In *Transactions on Graphics (TOG)*, volume 23, pages 309–314. ACM, 2004.

[16] Carsten Rother, Tom Minka, Andrew Blake, and Vladimir Kolmogorov. Cosegmentation of image pairs by histogram matching-incorporating a global constraint into mrfs. In *Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages 993–1000. IEEE, 2006.

[17] Lopamudra Mukherjee, Vikas Singh, and Charles R Dyer. Half-integrality based algorithms for cosegmentation of images. In *Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2028–2035. IEEE, 2009.

[18] Yuhang Zhang, Richard Hartley, John Mashford, and Stewart Burn. Superpixels via pseudo-boolean optimization. In *International Conference on Computer Vision (ICCV)*, pages 1387–1394. IEEE, 2011.

[19] Maxwell D Collins, Jia Xu, Leo Grady, and Vikas Singh. Random walks based multi-image segmentation: Quasiconvexity results and gpu-based solutions. In *Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1656–1663. IEEE, 2012.

[20] Edward Kim, Hongsheng Li, and Xiaolei Huang. A hierarchical image clustering cosegmentation framework. In *Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 686–693. Ieee, 2012.

[21] Andrew Y Ng, Michael I Jordan, Yair Weiss, et al. On spectral clustering: Analysis and an algorithm. In *Neural Information Processing Systems (NIPS)*, volume 14, pages 849–856, 2001.

[22] Sara Vicente, Carsten Rother, and Vladimir Kolmogorov. Object cosegmentation. In *Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2217–2224. IEEE, 2011.

[23] Fanman Meng, Hongliang Li, Guanghui Liu, and King Ngi Ngan. Object co-segmentation based on shortest path algorithm and saliency model. *IEEE Transactions on Multimedia*, 14(5):1429–1441, 2012.

[24] Linli Xu, James Neufeld, Bryce Larson, and Dale Schuurmans. Maximum margin clustering. In *Proceedings of Neural Information Processing Systems (NIPS)*, volume 17, pages 1537–1544, 2004.

[25] Yuri Y Boykov and M-P Jolly. Interactive graph cuts for optimal boundary & region segmentation of objects in nd images. In *International Conference on Computer Vision (ICCV)*, volume 1, pages 105–112. IEEE, 2001.

[26] Svetlana Lazebnik, Cordelia Schmid, and Jean Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 2, pages 2169–2178. IEEE, 2006.

[27] Alex Levinshtein, Adrian Stere, Kiriakos N Kutulakos, David J Fleet, Sven J Dickinson, and Kaleem Siddiqi. Turbopixels: Fast superpixels using geometric flows. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(12):2290–2297, 2009.

[28] Jure Leskovec, Andreas Krause, Carlos Guestrin, Christos Faloutsos, Jeanne VanBriesen, and Natalie Glance. Cost-effective outbreak detection in networks. In *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 420–429. ACM, 2007.

[29] Xiaojin Zhu, Andrew B Goldberg, Jurgen Van Gael, and David Andrzejewski. Improving diversity in ranking using absorbing random walks. In *North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 97–104, 2007.

[30] Roger A Horn and Charles R Johnson. *Matrix analysis*. Cambridge University Press, 2012.

[31] Leo Grady. Random walks for image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(11):1768–1783, 2006.

[32] Radhakrishna Achanta, Appu Shaji, Kevin Smith, Aurelien Lucchi, Pascal Fua, and Sabine Süsstrunk. Slic superpixels compared to state-of-the-art superpixel methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(11):2274–2282, 2012.

[33] Chuan Yang, Lihe Zhang, Huchuan Lu, Xiang Ruan, and Ming-Hsuan Yang. Saliency detection via graph-based manifold ranking. In *Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3166–3173, 2013.

[34] Josef Sivic, Andrew Zisserman, et al. Video google: A text retrieval approach to object matching in videos. In *International Conference on Computer Vision (ICCV)*, volume 2, pages 1470–1477, 2003.

[35] Dhruv Batra, Adarsh Kowdle, Devi Parikh, Jiebo Luo, and Tsuhan Chen. icoseg: Interactive co-segmentation with intelligent scribble guidance. In *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3169–3176. IEEE, 2010.

[36] D. Batra, A. Kowdle, D. Parikh, and T. Luo, J. Chen. icoseg database. `http://chenlab.ece.cornell.edu/projects/touch-coseg`. Accessed: 2017-01-27.

[37] G. Kim and E. P. Xing. Flickrmfc database. `http://www.cs.cmu.edu/~gunhee/r_mfc.html`. Accessed: 2017-01-27.

[38] Christoph H Lampert, Hannes Nickisch, and Stefan Harmeling. Learning to detect unseen object classes by between-class attribute transfer. In *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 951–958. IEEE, 2009.

[39] Mayank Lahiri, Chayant Tantipathananandh, Rosemary Warungu, Daniel I Rubenstein, and Tanya Y Berger-Wolf. Biometric animal databases from field photographs: Identification of individual zebra in the wild. In *Proceedings of the 1st ACM International Conference on Multimedia Retrieval*, page 6. ACM, 2011.

[40] Seung-Seok Choi, Sung-Hyuk Cha, and Charles C Tappert. A survey of binary similarity and distance measures. *Journal of Systemics, Cybernetics and Informatics*, 8(1):43–48, 2010.