Computer Vision and Image Understanding 154 (2017) 182–191

Contents lists available at ScienceDirect

Computer Vision and Image Understanding

journal homepage: www.elsevier.com/locate/cviu

Water detection through spatio-temporal invariant descriptors

Pascal Mettes[a,b,*], Robby T. Tan[a,c], Remco C. Veltkamp[a]

[a] Department of Information and Computing Sciences, Utrecht University, Utrecht, the Netherlands
[b] Intelligent Systems Lab Amsterdam, University of Amsterdam, Science Park 904, Amsterdam, the Netherlands
[c] Multimedia Technology and Design Programme, SIM University, Singapore

ARTICLE INFO

Article history:
Received 29 October 2015
Revised 26 January 2016
Accepted 8 April 2016
Available online 12 April 2016

Keywords:
Water detection
Spatio-temporal descriptors
Fourier analysis
Invariants
Markov random fields

ABSTRACT

In this work, we aim to segment and detect water in videos. Water detection is beneficial for appllications such as video search, outdoor surveillance, and systems such as unmanned ground vehicles and unmanned aerial vehicles. The specific problem, however, is less discussed compared to general texture recognition. Here, we analyze several motion properties of water. First, we describe a video preprocessing step, to increase invariance against water reflections and water colours. Second, we investigate the temporal and spatial properties of water and derive corresponding local descriptors. The descriptors are used to locally classify the presence of water and a binary water detection mask is generated through spatio-temporal Markov Random Field regularization of the local classifications. Third, we introduce the Video Water Database, containing several hours of water and non-water videos, to validate our algorithm. Experimental evaluation on the Video Water Database and the DynTex database indicates the effectiveness of the proposed algorithm, outperforming multiple algorithms for dynamic texture recognition and material recognition.

1. Introduction

The goal of this work is water detection in both natural and man-made environments from videos. Spatio-temporal water detection finds applications in unmanned ground and aerial systems (e.g. self-driving cars, and UAV's (van Gemert et al., 2014)), outdoor surveillance, video search, and wildlife search. These applications are highlighted in Fig. 1. To the best of our knowledge, related work focuses on texture recognition in general, and thus does not specifically explore the motion properties of water.

We focus on investigating the spatio-temporal motion properties of water. In biological studies, the visual properties of water have been investigated in order to understand the visual attractiveness of water in human and animal vision. From the work of Schwind (1991), it is known that water insects are attracted to the horizontal polarization caused by the reflections of water surfaces. This observation has for example been used to explain why certain insects lay eggs on highways (Kriska et al., 1998). In videos however, polarization information is not captured. Human observers are still experts at water detection without polarization information, indicating that water contains valuable spatio-temporal motion properties that can be exploited. Here, we investigate which spatio-temporal motion properties make water distinctive.

Current methods for automatic water detection can be divided into two categories: in specialized systems or as part of a broader recognition framework. In the broader fields of material recognition (Hu et al., 2011; Mettes et al., 2014b; Sharan et al., 2013) and dynamic texture recognition (Chan and Vasconcelos, 2008; Doretto et al., 2003; Fazekas et al., 2009; Zhao and Pietikäinen, 2007) water is one of the target classes. In these works, the objective is to minimize the miss-classification rate over all classes and as a result, the distinctive properties of water specifically are not investigated. Furthermore, the focus is generally on classification or segmentation, but not on the joint problem as posed here. On the other hand, water detection in specialized settings, such as autonomous driving (Rankin and Matthies, 2006) and in maritime settings (Smith et al., 2003), either make non-generalizable restrictions on the movement and orientation of cameras (Rankin and Matthies, 2006) or use auxiliary data sources in their measurements (Rathinam et al., 2007; Scherer et al., 2012; Smith et al., 2003). To address the limitations of related work with respect to water detection specifically, this work provides an investigation into the temporal and spatial behaviour of water scenes.

This work reports three contributions. (1) We introduce a video pre-processing step to remove background reflections and inherent water colours. (2) We introduce a hybrid spatial and temporal descriptor for local water classification. For the temporal

* Corresponding author. Tel.: +31634724306.
E-mail addresses: P.S.M.Mettes@uva.nl (P. Mettes), robbytan@unisim.edu.sg (R.T. Tan), R.C.Veltkamp@uu.nl (R.C. Veltkamp).

http://dx.doi.org/10.1016/j.cviu.2016.04.003
1077-3142/

(a) Surveillance. (b) Water hazards [11]. (c) Aerial water maps [13]. (d) Wildlife search [1].

Fig. 1. Visual examples of practical applications that benefit from water detection.

descriptor, we analyze the periodicity and regularity of local water patches and derive a descriptor that captures these elements. For the spatial descriptor, we advocate Local Binary Patterns and we investigate what makes them suitable for local water detection. (3) We introduce a new dataset, the Video Water Database, for experimental evaluation and to encourage research into this topic. The Video Water Database, further discussed in Section 5, along with the code used in the experimentation will be made publicly available to encourage further research into this topic.

This work extends an earlier investigation into this topic (Mettes et al., 2014a) in multiple aspects. An improvement is proposed in the pre-processing stage to deal with areas on the border of multiple objects of reflection, by modeling the density of pixel values over time. Also, further analysis is performed to investigate whether the hybrid descriptor is able to capture the spatial and temporal behaviour water ripples. In the experiments, we evaluate whether our method can generalize to water conditions and water types not seen during training. Lastly, another fusion of the temporal and spatial descriptor is evaluated.

The layout of the rest of this paper is as follows. In Section 2, an overview of water detection in related work is provided. Section 3 introduces the pre-processing step of the videos and the analysis of the local behaviour of water. This is followed by the discussion on local probabilistic classification and spatio-temporal regularization in Section 4. Finally, Section 5 provides the experimental evaluation of the algorithm and the paper is concluded in Section 6.

2. Related work

Given the lack of specific attention given to water detection, an overview is provided with respect to two broader recognition tasks: material recognition and dynamic texture recognition. Also, an overview of water localization in specialized systems is provided.

2.1. Water in material recognition

The classification of materials and static textures in images has a long history of investigation (Everts et al., 2012; Ojala et al., 2002; Sharan et al., 2013; Varma and Zisserman, 2009). Works on this topic are in line with Biederman (1987), who conjectured that materials are recognized in human vision by their surface characteristics such as texture and colour. Well-known approaches include filter bank distributions (Varma and Zisserman, 2005), Local Binary Patterns (Ojala et al., 2002), and image patch exemplars (Varma and Zisserman, 2009). In recent works, a shift has been made from laboratory settings (Ojala et al., 2002; Varma and Zisserman, 2005; 2009) to real-world image databases (Hu et al., 2011; Mettes et al., 2014b; Sharan et al., 2013). In these works, a range of surface characteristics, e.g. texture, colour, and reflectance, to find out what characteristics are best for classification. The results of these works indicate that spatial information is informative for distinguishing different materials. For water detection in videos however, there are two limiting aspects. First, only the spatial characteristics are investigated, excluding valuable temporal information. Second, research into material recognition has focused mostly on solving the classification problem or the segmentation problem, but not their joint problem.

2.2. Water in dynamic texture recognition

Dynamic textures are part of a class of motions with either structural or statistical similarity in both space and time (Nelson and Polana, 1992). Exemplary dynamic textures include fire, water, flags, and weather patterns. One of the dominant approaches in dynamic texture recognition is based on optical flow statistics (Chen et al., 2013; Fazekas et al., 2009; Fazekas and Chetverikov, 2007; Vidal and Ravichandran, 2005). In these approaches, either a global description is generated using invariant flow statistics such as characteristic direction and magnitude of flow vectors (Fazekas and Chetverikov, 2007), or flow vectors are binned into Histograms of Optical Flow (HOOF) (Chen et al., 2013). The use of optical flow is intuitively interesting for water detection, as the spatio-temporal movement of water seems statistically different to related textures. The use of conventional optical flow is however problematic for water detection in videos, as water meets none of the requirements for a proper flow estimation: Lambertian surface reflectance, pure translational motion parallel to the image plane, and uniform illumination (Beauchemin and Barron, 1995). A representation using optical flow will therefore be heavily influenced by the noise of the flow estimation, which makes optical flow not desirable for water detection.

Another popular research direction focuses on modeling dynamic textures as Linear Dynamical Systems (LDS) (Chan and Vasconcelos, 2008; Doretto et al., 2003; Ravichandran et al., 2013; Saisan et al., 2001). In dynamic texture recognition, the use of LDS has been made popular by Saison et al. (Saisan et al., 2001) and Doretto et al. (Doretto et al., 2003), mostly due to the proposed efficient sub-optimal learning procedure. As the original formulation of LDS requires a modeling of whole videos, it is unfit for local detection purposes. In order to deal with multiple textures within a video, several extensions have been provided. These include mixtures of dynamic textures (Chan and Vasconcelos, 2008), hierarchical EM clustering (Mumtaz et al., 2013), and Bags of Dynamical Systems (Ravichandran et al., 2013). These algorithms can potentially handle multiple textures in a video, but they have so far not been applied to detection problems. A noteworthy exception is the work of Ravichandran et al. (2011), where the joint segmentation and classification problem of dynamic textures is tackled by dividing a video into parts using Dynamic Appearance Images computed from LDS, after which the parts are represented by a bag-of-words representation with SIFT features. The representations are however more general and not tailored to water detection. Also, it is explicitly assumed that the texture class of a pixel does not change over time, restricting potential applications. Rather than performing a holistic modeling as with LDS, this work attempts to detect water from a local scale. The local scale is essential, as water is not bound to specific shapes in a scene.

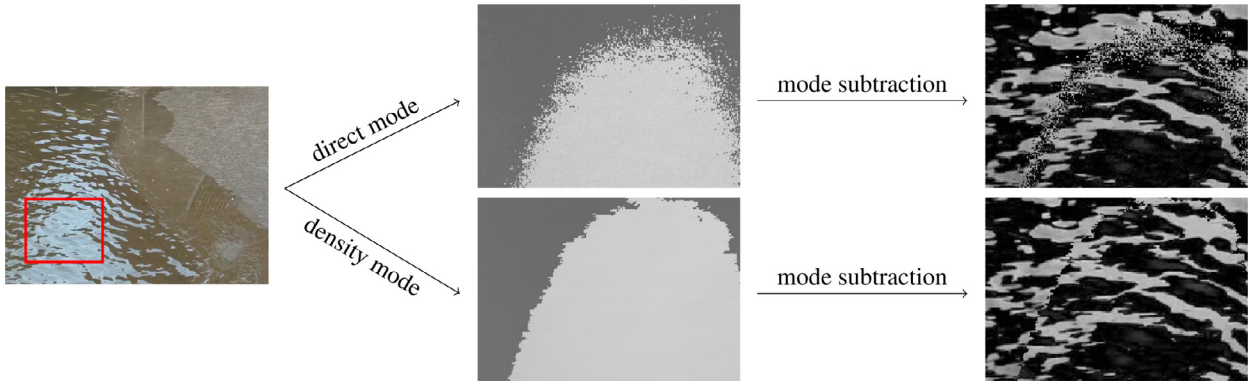


Fig. 2. Illustration of the effect of the proposed signal transformations. Left: a frame is shown for 3 water (blue) and 3 non-water (red) videos. Middle: a 2D projection of sampled local signals is shown for the water and non-water videos. Right: a 2D projection of the same signals after the transformations. Note that the signals of the water and non-water videos become nicely separated after the transformations, even in a 2D projection. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Notable is also the research on spatio-temporal Local Binary Patterns for dynamic texture classification (Zhao and Pietikäinen, 2007). In the work of (Zhao and Pietikäinen, 2007), both Volume LBP (VLBP) and LBP-TOP are introduced. VLBP generates longer histograms by adding binary comparison to temporal neighbours. Since the length of the histogram increases exponentially with the number of comparisons, VLBP typically yields histograms the size of 2^{14} or 2^{26}. More compact representations, three times the size of LBP, can be generated with LBP-TOP. Although VLBP and LBP-TOP can be extracted globally for a video, the length of their feature representation limits the applicability localized extraction. The lower dimensional spatial LBP remains interesting for water detection.

2.3. Water localization in specialized systems

Water detection has been investigated in specialized systems, including autoonomous driving systems (Iqbal et al., 2009; Rankin et al., 2010; Rankin and Matthies, 2006), maritime environments (Smith et al., 2003), and using flying robots (Rathinam et al., 2007; Scherer et al., 2012). Although these algorithms might provide a suitable solution in their restricted environment, none of the mentioned works are able to generalize to fully automatic water detection using minimally constrained video material.

In autonomous driving, several works have attempted to detect water hazards such as puddles and canals, to inform autonomous agents. In the work of Rankin and Matthies (2006), colour and texture cues are combined with stereo information to indicate water regions. Furthermore, estimated elevations are used to detect ground regions, decreasing the false positive rate. A similar method is introduced in Iqbal et al. (2009). A subsequent evaluation by Rankin et al. (2010) focuses on the specific scenario where stereo information is provided. In all works, additional sensors are used to help the detection. In maritime settings and in works using flying robots, similar non-generalizable assumptions have been made, whether it is assuming that water is within a specific part of the frame (Smith et al., 2003), requires a manual pre-processing step to identify sky regions (Scherer et al., 2012), or uses auxiliary sensors (Rathinam et al., 2007; Scherer et al., 2012). The works do therefore not generalize to water detection with minimal camera assumptions and without additional sensors, rendering them impractical for the problem of this work.

3. Local spatio-temporal water analysis

Since natural water scenes are dominated by aspects such as water colours, sky reflection, and object reflections, the videos are first pre-processed in a single-pass offline process. The pre-processing of a video results in a residual video, where these aspects are removed, to focus solely on water waves and ripples. After that, the temporal and spatial behaviour of water is analyzed, resulting in a novel temporal descriptor and the use of Local Binary Patterns (Ojala et al., 2002; Qian et al., 2011) as a spatial descriptor. By combining these descriptors into a hybrid descriptor, it becomes possible to locally detect water.

3.1. Residual videos

An important part in the process of detecting natural water scenes is dealing with the inherent variability of water, due to water colour, reflections, ripples, waves, and weather conditions. Instead of exploiting consistencies among these elements of variability, the focus of this work is to generate descriptions that are invariant to these variations.

To accommodate the temporal and spatial descriptor towards a distinctive water representation that is invariant to elements such as reflections and water colours, the videos in the database are first pre-processed. The goal of this step is to capture and subtract water reflections and colours from the video frames, yielding residual frames. The dominant reflections and colours are obtained for each pixel by computing the mode value over the frames. The underlying assumption is that water ripples and waves form a temporary disruption of an otherwise direct reflection. The most often occurring intensity values indicate the dominant water colours and reflections. As such, the temporal mode frame M of a video is defined pixel-wise as follows:

$$M_{\text{direct}}(x,y) = \arg\max_{i} \sum_{j=1}^{t} [\![I_j(x,y) = i]\!], \tag{1}$$

where t denotes the number of frames in the video, the Iverson brackets $[\![\cdot]\!]$ denote the indicator function, and $i \in \{0, .., 255\}$ denotes the set of intensity values. The residual frames can be obtained simply by means of absolute differencing the frames of the video with the temporal mode frame.

The use of the temporal mode for each pixel is not a stable choice under all circumstances. Most notably, as illustrated in Fig. 3, on strong edges, e.g. on the border of sky and object reflections, this approach yields a noisy result. To create more coherent residuals, Kernel Density Estimation (Scott, 2009) is performed over the set of intensity values for each pixel. In other words, for

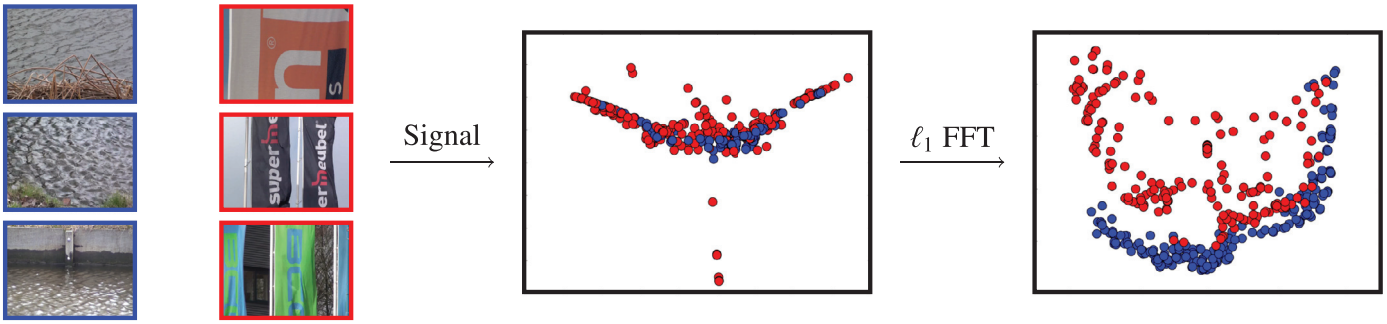


Fig. 3. The process of removing reflection highlights and water colours. Left: frame of a video containing water. Top middle and right: the temporal mode computed using Eq. 1 over the whole video and the corresponding residual of the frame, both for the selected red region. Bottom middle and right: the temporal mode computed using Eq. 2 and the corresponding residual. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

a pixel (x, y), the mode value is determined as

$$M_{\text{density}}(x,y) = \arg\max_i \frac{1}{t}\sum_{j=1}^{t} K_h(i - I_t(x,y)), \tag{2}$$

where $K_h(\cdot)$ denotes the Gaussian kernel and the bandwidth h is estimated using Scott's Rule (Scott, 2009). In Fig. 3, the mode frame using KDE is also shown for the example video. Contrary to the original method, this approach yields proper mode frames even at boundaries of two or more dominant colours.

The frames of the videos are downsized to a quarter of their width and height, in order to remain computationally practical. This is since the use of Gaussian KDE over Eq. 2 increases the time complexity of computing the mode frame of a video from $O(p(t+i))$ to $O(pti)$, with p the number of pixels, t the number of frames, and i the number of intensity values.

3.2. Local temporal water behaviour

For the temporal descriptor, a Eulerian approach is opted over a Lagrangian. In other words, rather than tracking pixels over time as is done with optical flow (the Lagrangian approach), the dynamics of water is investigated from static locations. The hypothesis behind this is that transitions of brightness values over time contain valuable information regarding the characteristics of water. It is hypothesized that they include gradual motion (waves enter and exit a local area smoothly), repetitive motion (waves re-occur in similar fashion over time), and regular motion (waves re-occur at similar intervals).

As the brightness transitions of individual pixels are sensitive to noise, the local temporal behaviour of water is analyzed by averaging brightness values of a local region around a pixel. For a spatio-temporal video volume, an m-dimensional signal is generated by computing the mean brightness value of an $n \times n$ patch around a pixel for m consecutive frames. Note that this is similar to a 3D mean convolution filter of size $n \times n \times m$. The resulting list of brightness values can be seen as a signal. These signals exhibit more sinusoidal patterns when extracted locally from water regions than from non-water regions, as is expected from the hypothesized motion characteristics of water.

Using the signals obtained from the local 3D convolution, the primary concern becomes finding a descriptor that generates a small distance between two water signals and a large distance between a water and non-water signal with respect to the hypotheses. An obvious solution is to directly compute the ℓ_2 distance between two signals, i.e. to directly use the signals as the temporal water descriptor. This solution is however erroneous, as the signals lack a number of invariance properties. A descriptor based on the m-dimensional signals should in effect be invariant to temporal shifts, brightness shifts, and brightness amplitudes. This can be generated by computing the minimum distance between two signals S_1 and S_2 under all temporal shifts T, brightness shifts B, and amplitudes A

$$d(S_1,S_2) = \min_{t\in T, b\in B, a\in A} \sum_{i=1}^{m} S_1[i] - a\cdot(S_2[i+t]+b). \tag{3}$$

The above equation is however prohibitively expensive. A more scalable approach is to create temporal and brightness shift invariance by computing the Fast Fourier Transform (FFT). For an m-dimensional signal S, the m-dimensional Fourier transform F is computed as follows:

$$F_i = |\sum_{j=1}^{m} S_j \exp(-2\pi i j\sqrt{-1}m)|. \tag{4}$$

Note that in Eq. 4, the variable i does not denote the imaginary number, but the index of the Fourier transform; the imaginary number is for convenience written explicitly as $\sqrt{-1}$.

Computing distances between two Fourier signals creates (temporal and brightness) shift invariance in $O(m\log m)$ time. However, since the descriptor is not invariant against amplitudes, the final temporal descriptor is generated by performing ℓ_1 normalization. An invariance with respect to brightness amplitudes is desirable, as two descriptors with similar levels of regularity and repetition will have a large distance in both the original signal space and the Fourier transform space if their amplitudes are not similar (e.g. rough and calm water). Using the ℓ_1 normalization to add the final layer of invariance, a temporal water descriptor $\{F_i\}_{i=1}^m$ is computed from an original signal $\{S_i\}_{i=1}^m$ as

$$F_i = \frac{|\sum_{j=1}^{m} S_j \exp(-2\pi i j\sqrt{-1}m)|}{\sum_{k=1}^{m}|\sum_{j=1}^{m} S_j \exp(-2\pi k j\sqrt{-1}m)|}. \tag{5}$$

A practical justification of adding the layers of invariance is provided in Fig. 2. In the example of the Figure, signals are randomly sampled from 3 water and 3 flag videos. In the 2D projection (Tenenbaum et al., 2000) of the original signals, the water and flag signals are completely indistinguishable. After adding the desired elements of invariance, the signals become nicely separable, even in a 2D projection of 200D descriptors.

3.3. Local spatial water behaviour

Although the above introduced water descriptor can capture the temporal behaviour of water, it explicitly ignores the spatial layout of water waves and ripples. Due to the deformable nature of water, a descriptor is desired that can provide spatial information without explicitly modeling water waves and ripples. To meet this desire, Local Binary Pattern histograms (Ojala et al., 2002; Qian et al., 2011) are investigated. LBP histograms have a number of benefits

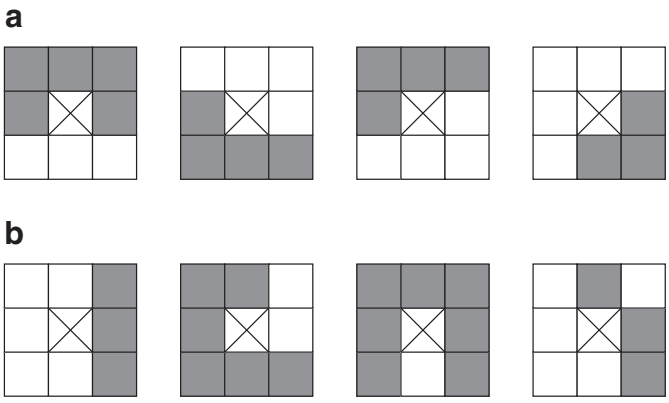


Fig. 4. Illustration of LBP values that (a) correlate positively to water and (b) correlate negatively to water. White boxes indicate a value of 1. Note how positively correlated LBP values have a more equal ratio between 1 and 0 values and are typically uniform (i.e. contain only 2 transitions between 1 and 0 values across the 8 connected neighbours).

particularly desired properties for the purpose of this work. First and foremost, the spatial arrangement of individual pixels only extends to a one pixel neighbourhood. This is convenient, because of the deformations possible within a patch. On the other hand, the histograms are of sufficient dimensionality to be discriminative.

The LBP value of a single pixel is computed by comparing the intensity value of the pixel to nearby pixels. Here, the LBP-variant using the 8 direct neighbours is explored, i.e. the LBP-value of a pixel is determined as

$$\mathrm{LBP}(g^c) = \sum_{p=0}^{7} [\![g_p^c - g^c \geq 0]\!] \cdot 2^p, \tag{6}$$

where g^c denotes the pixel to be evaluated and $\{g_p^c\}_{p=0}^7$ denotes the 8 direct neighbours. In order to compute a descriptor over a local region, the LBP-value is computed for each pixel in that region and placed in one of the $2^8 = 256$ bins, according to the value yielded by Eq. 6.

The question remains whether the use of LBP histograms is beneficial for water specifically. To investigate this, a number of local patches of water and non-water videos are randomly extracted and trained using a linear classifier, in this case a linear Support Vector Machine. In Fig. 4, we show LBP values for which the corresponding SVM weights show resp. positive and negative correlation with respect to water. From the Figure, it can readily be observed that positive LBP values fire on the edges of ripples. Furthermore, the ratio between 1 and 0 values for positively correlated LBP values is more equal and for negatively correlated LBP values. A substantial part of the LBP values with more than two transitions (similar to the rightmost example of Fig. 4b) are to some extend negatively correlated to water, indicating that the spatial layout of water waves and ripples is not chaotic and is characterized by smooth spatial transitions.

Note that the use of Local Binary Patterns results in a descriptor aimed at extracting gradient information. Throughout this work, it is however referred to as a spatial descriptor to exemplify the contrast to the temporal descriptor; the temporal descriptor tries to identify patterns in the temporal dimension, the spatial descriptor does the same in the two spatial dimensions.

4. Classification and regularization

Given the pre-processed videos and a local temporal and spatial descriptor, the final goal becomes generating a detection mask for each frame of a video. This is performed in two steps; direct probabilistic classification and spatio-temporal regularization. In the first step, a model is created from positively and negatively sampled descriptors. This model can then be applied to a test video, resulting in a large number of independent classifications. These classifications can already be served as detections by binarizing all the probabilities. As the learned model does not perfectly classify each local video volume, the output of the individual classifications is noisy and a form of regularization is required to generate coherent water detection masks.

The derived temporal and spatial descriptors are not used as local features for a more global encoding; rather, a model is generated directly from individual descriptors. In this work, both the early and late fusion variant are experimentally evaluated (Snoek et al., 2005). In early fusion, the temporal and spatial descriptors computed from a local video volume are first concatenated, after which a model is trained on these hybrid descriptors. Contrarily, in late fusion, a model is trained separately for the temporal and spatial descriptors, and the probability of a local video volume of being water is determined by averaging the scores from the two models.

4.1. Local probabilistic classification

Classification is performed by sampling local descriptors from training videos. These descriptors are then used as feature vectors for the training of the classifier, where the labels are inherited from the video from which they were sampled. As the total number of local video patches and volumes over all training videos is cumbersomely large, a random sampling approach is adopted here. To maximize coverage, each frame of each training video is evaluated, and a low number of descriptors are extracted from randomly sampled locations.

The yielded set of feature vectors are then fed to a Random Decision Forest (Criminisi et al., 2012). The use of a Decision Forest is particularly interesting, since it provides probabilistic outputs and it inherently generates non-linear decision boundaries. Probabilistic outputs will prove to be useful, as the uncertainty can be used for regularizing the detection. Local descriptors are extracted and independently classified for each frame of a test video.

4.2. Spatio-temporal regularization

The procedure of Section 4.1 generates hundreds of individual local water probabilities per frame of a test video. As the classification procedure is not expected to be fully accurate, a number of miss-classifications are bound to occur, even within a single frame. The additional information gained by computing probabilities over binary labels opens up the possibility to handle classification outliers. Under the intuitive assumption that water regions have a high spatial support (i.e. there are a limited number of boundaries between water and non-water regions), a final detection map per frame of a video can be computed by means of regularization. Here, the regularization takes of form of a binary Markov Random Field (Boykov and Kolmogorov, 2004), that attempts to solve the following minimization objective:

$$f(x,y) = \sum_{p \in V} V_p(x_p) + \lambda \sum_{(p,q) \in C} V_{pq}(x_p, x_q), \tag{7}$$

where the unitary term V_p denotes the match between the label of node p and its corresponding probability

$$V_p(x_p) = \begin{cases} 1 - P_p & \text{if } x_p \text{ is water} \\ P_p & \text{otherwise,} \end{cases} \tag{8}$$

where P_p denotes the probability of node p of being water. The pairwise term V_{pq} of Eq. 7 follows the well-known 0/1 Potts model that enforces similarity between the labels of nodes from the same

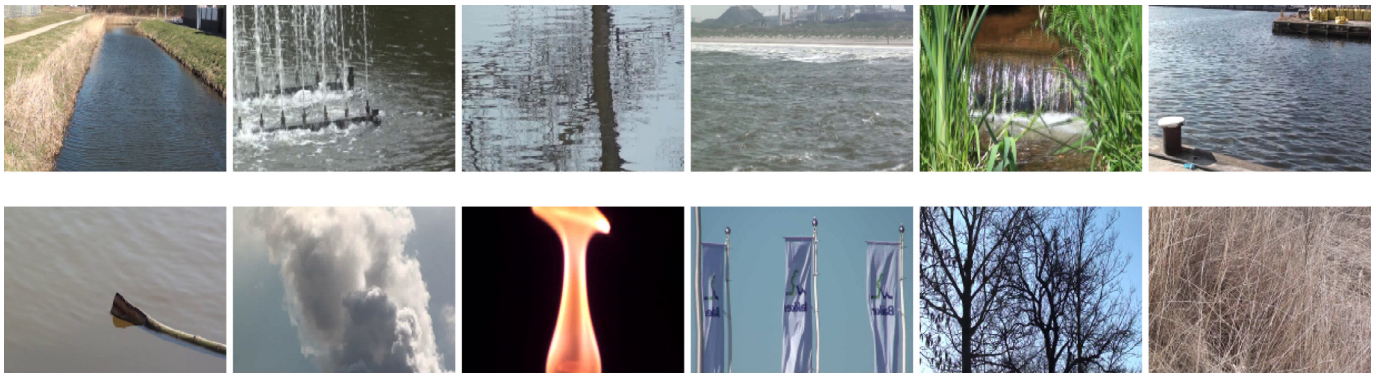


Fig. 5. An example frame of each of the subcategories present in the Video Water Database. See text for details on the subcategories.

clique (Boykov and Kolmogorov, 2004). The term λ is a hyperparameter weighting the importance between the unitary and pairwise terms.

An obvious choice of cliques in the Markov Random Field are the pairwise spatial neighbours within a single frame of a video. This would involve generating a single Markov Random Field for each frame. As it is furthermore desired to penalize different labellings at the same location between consecutive frames, the pairwise temporal neighbours are also used as cliques. This results in a single spatio-temporal Markov Random Field for each evaluated video.

5. Experimentation

To validate the proposed algorithm for water detection, the algorithm is evaluated on two different but related tasks; water detection and water classification-by-selection.

The detection quality of a video is defined as the average of the fit of the detection fit per frame. Formally, the detection fit D of a binarized video V compared to a ground truth mask M is defined as

$$D(V, M) = \frac{\sum_{i=1}^{|V|} d(V_i, M)}{|V|}, \tag{9}$$

where $|V|$ denotes the number of frames in V, V_i denotes the i^{th} frame, and $d(V_i, M)$ is defined as

$$d(V_i, M) = 1 - \frac{\sum_{x=1}^{W} \sum_{y=1}^{H} |V_i[x, y] - m[x, y]|}{W \cdot H}, \tag{10}$$

where W and H denote the width and height of the frame and the pixel values of the computed detection and the mask m are 1 for water and 0 otherwise.

The classification-by-selection task is a more lenient task; given a selected area in a video, determine whether that area is a water surface or not. Although not as informative as the detection task, this task does offer several insights; it serves its own set of applications, such as human-aided water detection (i.e. water detection where the user specifies an interesting region). Also, it opens up the possibility for comparison against works from fields such as material and dynamic texture recognition.

5.1. The video water database

Due to the lack of attention given to the specific task of water detection in videos, no database is available with a large enough quantity and variety for desirable evaluation. Therefore, the Video Water Database (VWD) is introduced here. This database contains several hours of video material of a wide range of water and non-water scenes. To the best of our knowledge, this is the largest database with video material on water.

In total, the database consists of 260 videos, where each video contains between 750 and 1500 frames, all with a frame size of 800 $\times$ 600. The water class consists of 160 videos of predominantly 7 subcategories; canals, fountains, lakes, oceans, ponds, rivers, and streams. The non-water class can be represented by any other scene. Here, the non-water class contains subcategories with similar spatial and temporal characteristics; clouds/steam, fire, flags, trees, and vegetation. An example of each of the subcategories in the database is shown in Fig. 5. All the videos are taken with a static camera, i.e. there are no large camera motions. Static cameras are employed here to be able to investigate the temporal and spatial properties of water in isolation. It furthermore allows us to quantify the performance of our hybrid descriptor. In order to compute the quality of the computed detections, a binary mask is created for each video stating which pixels are water and which pixels are not. Care has furthermore been taken to maintain a large variety in scale and orientation of the water and non-water surfaces.

We note that our algorithm does not explicitly assume a static camera. Both the pre-processing and the temporal descriptor assume a temporal window at a specific spatial location. The temporal window in turn forms a trade-off; a smaller temporal window increases the robustness to camera motion, at the cost of discriminative power.

Besides evaluating on the Video Water Database, a subset of 75 videos from the DynTex database (Péteri et al., 2010) is also used for evaluation. The motive for this evaluation is two-fold. First, it shows that the algorithm is not tailored to the created database. Second, it provides a comparison for water detection against other non-water textures and objects. The selected subset contains humans, animals, traffic, windmills, flowers, and cloths. Since most of the named textures and objects will not be seen during training, the effectiveness of the algorithm on the DynTex database will provide insight into the generalization properties to unseen negatives.

5.2. Implementation details

For the temporal descriptor, the length of the signal constitutes a trade-off between discriminative prowess and practicality. As the focus here is on detection and accuracy, a signal length of $m = 200$ is used, with a resulting 200D feature vector. When combining the temporal and spatial descriptors before classification, the 200D temporal descriptor and 256D spatial descriptor are simply concatenated.

During training, 10 samples are retrieved from random locations for each frame of each training video, yielding roughly 750.000 samples to be trained by the Decision Forest. The main parameters of the Forest – the randomness and the number of

Table 1
Overview of the detection results of the descriptors and their fusions, resp. without and with regularization.

	No regularization			
	Temporal	*Spatial*	*Late fusion*	*Early fusion*
Water	90.4	85.5	90.7	**90.8**
Non-water	67.3	86.7	85.9	**92.1**
Average	78.9	86.1	88.3	**91.5**
	Spatio-temporal regularization			
	Temporal	*Spatial*	*Late fusion*	*Early fusion*
Water	92.0	87.1	91.4	**92.3**
Non-water	68.6	89.8	90.7	**95.0**
Average	80.3	88.4	91.1	**93.7**

trees – are set through validation (Van Gemert et al., 2009). For a test video, samples are extracted every 11^{th} pixel in width and height for each frame, followed by individual classification. For the regularization, an equal contribution of the unary and pairwise terms (i.e. $\lambda = 1$) has empirically shown to be most effective.

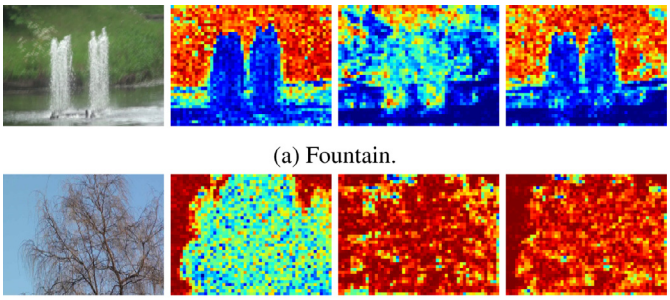
To evaluate the effectiveness of the algorithm for water detection, the primary components – temporal and spatial features extraction, fusion, regularization – are evaluated on the newly introduced Video Water Database. For this, the database is split randomly with equal ratio into a train and test split; equally among all subcategories.

5.3. Water detection in the video water database

In Table 1, an overview is provided of the detection results for the algorithm without and with regularization. Individually and without regularization, the temporal and spatial descriptors yield 78.9% and 86.1% detection accuracy. It is interesting to observe that the water descriptor yields good performance for water, while the spatial descriptor yields good performance for non-water. The complementary nature also comes back in the performance of the fusions of the descriptors. The best performance is achieved by performing early fusion, with an increase to 91.5% average detection rate. Early fusion is preferred here because of the large difference in true and false rates of the individual descriptors. In late fusion, the mistakes of an individual descriptor greatly influences the final detection result (due to the equal weighting of the probabilities). Early fusion however makes it possible to compensate for each other's mistakes during training.

Next to fusing temporal and spatial information for local water classification, Table 1 also indicates the effectiveness of applying regularization. Enforcing label consistency among spatio-temporal cliques removes classifier outliers and results in a smooth final detection result. The combination of the hybrid descriptor and spatio-temporal regularization yields a final detection accuracy of 93.7%. In Fig. 7, the final detection result is shown for a number of test videos.

Interestingly, the strong increase in performance for the hybrid descriptor is not because of a strong increase in true detection rate. Contrarily, it is the false detection rate that achieves a strong decrease; from 32.7% (temporal) and 13.3% (spatial) to 5%. This indicates that non-water elements might resemble water temporally or spatially, but not always spatio-temporally. The reasoning behind this observation is for a substantial part captured in Fig. 6. In this Figure, two frames of test videos are shown, as well as the probability maps. In the probability maps, a blue colour indicates water, while a red colour indicates non-water. In Fig. 6a, it is shown that the temporal descriptor can aid the spatial descriptor, while Fig. 6b shows that the spatial descriptor can aid the temporal descriptor.



(a) Fountain.

(b) Tree.

Fig. 6. Two frames indicating the complementary nature of the temporal and spatial information. The first column shows a frame of the video. The second column shows the probability map of the temporal descriptor, the third column shows the map for the spatial descriptor, and the fourth column shows the map for the hybrid descriptor.

Table 2
Overview of the detection results on the DynTex database subset. The numbers within parentheses represent the detection results without regularization.

	Temporal	*Spatial*	*Hybrid*
Water	89.7 (87.5)	70.0 (68.4)	**87.9** (83.0)
Non-water	63.7 (64.3)	85.7 (79.6)	**94.7** (89.5)
Average	76.7 (75.9)	77.9 (74.0)	**91.3** (86.2)

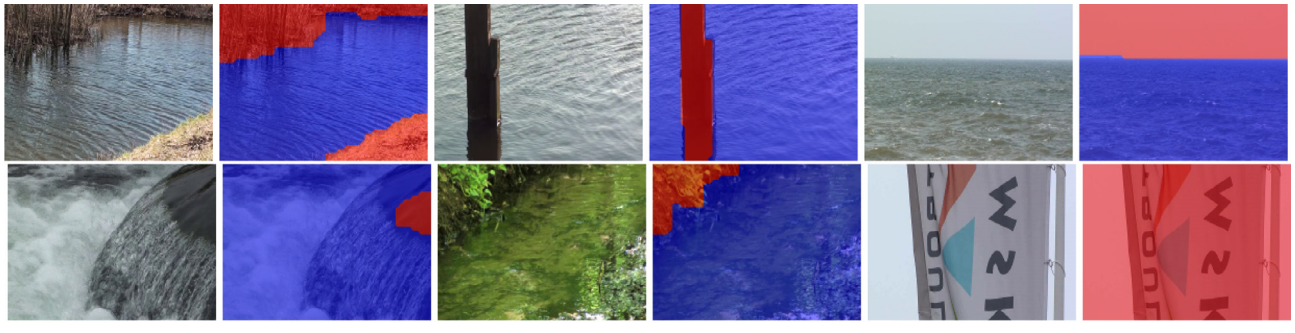
To iterate the effectiveness of the temporal descriptor for water detection, we evaluate the effectiveness of the temporal descriptor as a function of the length of its descriptor. In Fig. 8, we show the detection accuracy for water and non-water as a function of the descriptor length. Interestingly, the Figure shows that the detection of water is hardly affected by the length of the signal, which means that our temporal descriptor can capture characteristic temporal patterns of water with a short exposure. On the other hand, non-water accurary is highly affected by the length of the signal. This result indicates that our temporal descriptor is highly tailored to water detection, rather than focused on the general case of (dynamic) texture recognition.

5.4. Water detection in the DynTex database

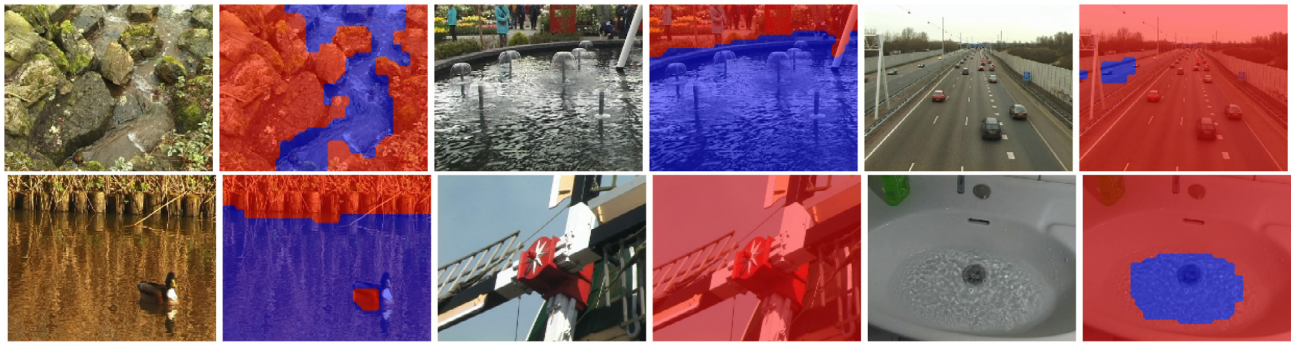
To further emphasize the effectiveness of the introduced algorithm and in order to investigate the generalization properties of the algorithm, the water detection is also performed on a subset of the DynTex database (Péteri et al., 2010). In total, 75 videos of water and non-water scenes are selected. For training the model, the train split of the Video Water Database is used, while the model is evaluated on all the selected videos from the DynTex database.

Since the videos form the DynTex database have been captured with a different intent than the Video Water Database, different water and non-water types are present in this subset. For water, elements such as drinking water and water surfaces during rainfall are present. For non-water, new elements include windmills, animals, humans, and traffic. As these elements are not present in the training videos of the Video Water Database, a proper detection and classification of these videos greatly depends on the generalizing properties of the algorithm.

In Table 2, an overview is provided of the detection accuracies yielded on the DynTex subset. Although the numbers of Table 1 and 2 are not directly comparable, the comparison does provide an indication of the generalization properties of the algorithm. Individually, the descriptors yield a lower detection accuracy on the DynTex database subset. However, the early fusion into the hybrid water descriptor results in a substantial boost from 76.7%



(a) Video Water Database.



(b) DynTex database.

Fig. 7. Examples of detection results shown for both databases. Blue indicates water, red indicates non-water.

(temporal) and 77.9% (spatial) to 91.3% detection accuracy on average.

Fig. 7 b shows exemplary detections on the DynTex subset. For multiple examples, no similar video is present in the training set, e.g. the drinking water in the sink, the windmill, and the traffic. This Figure paints a similar picture to the results of Table 2; the algorithm can generalize to previously unseen water and non-water subcategories. This result highlights the goal of the algorithm to capture the inherent properties of water.

5.5. Water classification-by-selection

As a proof of concept and in order to compare the algorithm to a number of related papers, binary water classification is also considered. Here, the goal is to determine whether a video supplemented with a binary mask is water or not. The same training and testing splits are used as the detection task, while the manually created binary masks serve as binary masks to determine the foreground region.

For the introduced algorithm, the classification of a video is a function of the ratio of water and non-water pixels in the foreground region. Agnostic to any prior on the ratio of water and non-water pixels, a video is classified as water if the ratio of water pixels is at least $\frac{1}{2}$, otherwise it is classified as non-water.

The classification accuracy of the algorithm is compared to multiple generic baselines from material and dynamic texture classification. In total, 6 algorithms are used as baseline methods. These baselines serve as general indicators of the complexity of the problem. The baseline algorithms include Transferred ConvNet Feature (Qi et al., 2016), Volume Local Binary Patterns (Zhao and Pietikäinen, 2007), LBP-TOP (Zhao and Pietikäinen, 2007), Linear Dynamical Systems (Doretto et al., 2003), Gabor filter bank distributions (Varma and Zisserman, 2005), and optical flow statistics (Fazekas and Chetverikov, 2007).

For Volume LBP (Zhao and Pietikäinen, 2007), a 2^{14}-dimensional feature vector is generated for a video by means of histogram binning using the 14 direct temporal and spatial neighbours of sampled foreground pixels. For LBP-TOP (Zhao and Pietikäinen, 2007), a $(3{\cdot}2^8)$-dimensional feature vector is generation by histogram binning the 8 neighbours of the 3 orthogonal planes of sampled foreground pixels. Both VLBP and LBP-TOP are trained and tested using nearest neighbour classification in Euclidean space (Zhao and Pietikäinen, 2007).

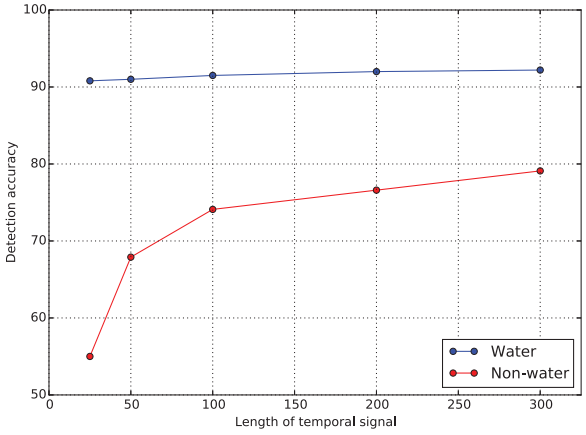


Fig. 8. Detection accuracy as a function of the length of the temporal descriptor. Note that a only short exposure is required for high water detection accuracy.

For LDS (Doretto et al., 2003), the whole video has to be used, as the number of pixels needs to match between a pair of videos. Here, we follow the setup of Saisan et al. (2001) and model each video in the space of dynamical systems. A distance measure is defined by the Martin distance (Saisan et al., 2001). We have employed the Dynamic Texture Toolbox for this baseline.[1] For the MR8 filter bank (Varma and Zisserman, 2005), we comute

[1] http://cis.jhu.edu/~avinash/projects/DTBox/

Table 3
Classification accuracy results yielded for both the Video Water Database (second column) and the Dyntex database (third column). The fourth column states the absolute difference in achieved accuracy between the Video Water Database and the Dyntex database.

Methods	*VWD*	*Dyntex*	*Abs. diff.*
Ours, hybrid	**98.4**	**95.8**	-2.6
Ours, spatial	93.1	84.6	-8.5
Ours, temporal	83.0	81.0	-2.0
st-TCoF (Qi et al., 2016)	97.2	90.0	-7.2
LBP-TOP (Zhao and Pietikäinen, 2007)	93.3	87.5	-5.8
Volume LBP (Zhao and Pietikäinen, 2007)	93.8	79.1	-14.7
MR8 filter bank (Varma and Zisserman, 2005)	84.3	67.2	-17.1
Flow stats (HS) (Fazekas and Chetverikov, 2007; Horn and Schunck, 1981)	75.0	55.4	-19.6
LDS (Doretto et al., 2003)	67.4	56.3	-11.1
Flow stats (LK) (Fazekas and Chetverikov, 2007; Lucas and Kanade, 1981)	62.8	49.7	-13.1

local 8-dimensional features from 38 filter responses by taking the maximum response over the rotations of different anisotropic filters (Varma and Zisserman, 2005). We construct a codebook of 250 clusters using k-means and represent each video using a 250-dimensional bag-of-words representation. Classification is performed using nearest neighbours with the χ^2 distance. For the optical flow, 4 flow statistics are computed on estimated flows and averaged over the video. These statistics include characteristic direction, characteristic magnitude, divergence, and curl (Fazekas and Chetverikov, 2007). The optical flow baseline is performed both using the flow algorithm of Lucas and Kanade (1981) and using the flow algorithm of Horn and Schunck (1981). Similar to VLBP and LBP-TOP, nearest neighbours in Euclidean space is employed here.

Besides comparing against hand-crafted features, we also compare against the deep convolutional video representation of Qi et al. (2016). For a video, we extract a 4096-dimensional representation using a pre-trained AlexNet (Krizhevsky et al., 2012). We compute the spatial and temporal TCoF (Transferred ConvNet Feature) of a video using the first and second order statistics. We ℓ_2 normalize the spatial and temporal TCoF, after which they are concatenated. This results in a 16,384-dimensional representation for each video, which is trained and tested using a Linear SVM (with C=40 (Qi et al., 2016)).

An overview of the classification accuracies is highlighted in Table 3. On the Video Water Database, the hybrid descriptor outperforms both the individual descriptors (similar to water detection) and the baseline methods. In fact, the only baseline method that comes near the results of the hybrid descriptor is TCoF (Qi et al., 2016). This result is not entirely surprising, given the strength of the deep representations from convolutional neural networks. However, our hybrid approach still outperforms TCoF for water detection specifically, highlighting its effectiveness.

As indicated in the third column of Table 3, all methods yield a lower classification accuracy on the Dyntex database. Although the numbers can not directly be compared to the numbers of the Video Water Database, the decline in performance of each of the methods provides a clear indication of the performance of the water algorithm. For the water algorithm, the hybrid and temporal descriptor indicate the best generalization capabilities, while the spatial descriptor reports a 8.5% decline (absolute difference). For the baseline methods, an overview higher decline is reported (between 6.5% and 19.6%). This result indicates that the introduced water algorithm not only outperforms the baseline methods, it is also able to generalize better to unseen water and non-water subcategories.

Both for the detection and classification tasks, it can be noted that the scores of respectively Table 2 and 3 are rather high. This is first and foremost due to the nature of the task; it is cast as a strictly binary problem. This means that if a local video volume or even a whole video of a tree is classified as fire, there will be no loss. As long as the water/non-water boundary line is not crossed, no loss occurs. Note however that this hardly makes the problem easy, especially from a purely local perspective. When treating each local video volume independently for classification, any form of contextual information is discarded. This is highlighted in Fig. 9. When looking at the whole frames, it is not hard to make out which one is water and which one is a cloud. However, purely based on the local squares, it becomes exceedingly harder to state which one is part of a water surface and which one is not. This indicates the complexity of a non-holistic approach to the detection problem.

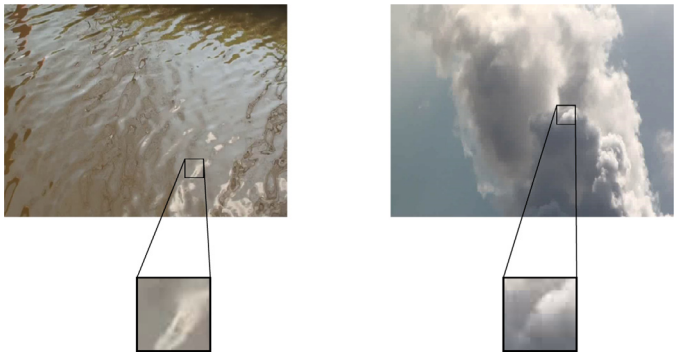


Fig. 9. Visual example indicating the complexity of water detection purely from local information.

6. Conclusions

In this work, the problem of detecting water in videos is tackled. As the specific problem of water detection has hardly been addressed in related work, this work investigates the temporal and spatial dynamics of water. First, a pre-processing stage is introduced that is aimed at removing reflections and water colours. After that, a hybrid descriptor and local detection algorithm are introduced for discovering water regions in a video. To evaluate the algorithm, the Video Water Database is furthermore introduced. Quantitative and qualitative evaluation show that the algorithm is able to robustly detect region of water in videos, with a high detection accuracy and a classification accuracy that outperforms algorithms from directly related fields.

Acknowledgement

This work is supported by the FES project COMMIT.

References

Beauchemin, S., Barron, J., 1995. The computation of optical flow. ACM Comput. Surveys 27 (3), 433–466.

Biederman, I., 1987. Recognition-by-components: a theory of human image understanding. Psychol. Review 94 (2), 115.

Boykov, Y., Kolmogorov, V., 2004. An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. Pattern Anal. Mach. Intell.

Chan, A., Vasconcelos, N., 2008. Modeling, clustering, and segmenting video with mixtures of dynamic textures. Pattern Anal. Mach. Intell. 30 (5), 909–926.

Chen, J., Zhao, G., Salo, M., Rahtu, E., Pietikainen, M., 2013. Automatic dynamic texture segmentation using local descriptors and optical flow. Trans. Image Process. 22 (1), 326–339.

Criminisi, A., Shotton, J., Konukoglu, E., 2012. Decision forests. Found. Trends Comput. Graph. Vision 7 (2), 81–227.

Doretto, G., Chiuso, A., Wu, Y., Soatto, S., 2003. Dynamic textures. Int. J. Comput. Vision 51 (2), 91–109.

Everts, I., Van Gemert, J.C., Gevers, T., 2012. Per-patch descriptor selection using surface and scene properties. In: Computer Vision–ECCV 2012. Springer, pp. 172–186.

Fazekas, S., Amiaz, T., Chetverikov, D., Kiyyati, N., 2009. Dynamic texture detection based on motion analysis. Int. J. Comput. Vision 82 (1), 25–32.

Fazekas, S., Chetverikov, D., 2007. Analysis and performance evaluation of optical flow features for dynamic texture recognition. Signal Process.: Image Commun. 22, 680–691.

van Gemert, J.C., Verschoor, C.R., Mettes, P., Epema, K., Koh, L.P., Wich, S., 2014. Nature conservation drones for automatic localization and counting of animals. In: European Conference on Computer Vision workshop (ECCVw).

Horn, B., Schunck, B., 1981. Determining optical flow. Artificial Int. 17 (1), 185–203.

Hu, D., Bo, L., Ren, X., 2011. Toward robust material recognition for everyday objects. Brit. Mach. Vision Conf. 48.1–48.11.

Iqbal, M., Morel, O., Meriaudeau, F., Komputer, F.I., 2009. A survey on outdoor water hazard detection. Information and Communication Technology and Systems.

Kriska, G., Horváth, G., Andrikovics, S., 1998. Why do mayflies lay their eggs en masse on dry asphalt roads? water-imitating polarized light reflected from asphalt attracts ephemeroptera.. J. Exp. Biol. 201 (15), 2273–2286.

Krizhevsky, A., Sutskever, I., Hinton, G.E., 2012. Imagenet classification with deep convolutional neural networks. NIPS.

Lucas, B., Kanade, T., 1981. An iterative image registration technique with an application to stereo vision. Int. Joint Conf. Artificial Intell. 674–679.

Mettes, P., Tan, R., Veltkamp, R., 2014. On the segmentation and classification of water in videos. In: VISAPP 2014 - Proceedings of the 9th International Conference on Computer Vision Theory and Applications, Volume 1, Lisbon, Portugal, 5-8 January, 2014, pp. 283–292. doi:10.5220/0004680202830292.

Mettes, P., Tan, R.T., Veltkamp, R.C., 2014. A bottom-up approach to class-dependent feature selection for material classification. In: VISAPP 2014 - Proceedings of the 9th International Conference on Computer Vision Theory and Applications, Volume 2, Lisbon, Portugal, 5-8 January, 2014, pp. 494–501. doi:10.5220/0004721204940501.

Mumtaz, A., Coviello, E., Lanckriet, G., Chan, A., 2013. Clustering dynamic textures with the hierarchical em algorithm for modeling video. Pattern Anal. Mach. Intell. 35 (7), 1606–1621.

Nelson, R., Polana, R., 1992. Qualitative recognition of motion using temporal texture. Comput. Vision Image Understand. 56 (1), 78–89.

Ojala, T., Pietikainen, M., Maenpaa, T., 2002. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. Pattern Anal. Mach. Intell. 24 (7), 971–987.

Péteri, R., Fazekas, S., Huiskes, M., 2010. Dyntex: A comprehensive database of dynamic textures. Pattern Recogn. Lett. 31 (12), 1627–1632.

Qi, X., Li, C.-G., Zhao, G., Hong, X., Pietikäinen, M., 2016. Dynamic texture and scene classification by transferring deep image features. Neurocomputing 171, 1230–1241.

Qian, X., Hua, X.-S., Chen, P., Ke, L., 2011. Plbp: An effective local binary patterns texture descriptor with pyramid representation. Pattern Recogn. 44 (10), 2502–2515.

Rankin, A., Ivanov, T., Brennan, S., 2010. Evaluating the performance of unmanned ground vehicle water detection. In: Proceedings of the 10th Performance Metrics for Intelligent Systems Workshop. ACM, pp. 305–311.

Rankin, A., Matthies, L., 2006. Daytime water detection and localization for unmanned ground vehicle autonomous navigation. Proceed. 25th Army Science Conf.

Rathinam, S., Almeida, P., Kim, Z., Jackson, S., Tinka, A., Grossman, W., Sengupta, R., 2007. Autonomous searching and tracking of a river using an uav. In: American Control Conference. IEEE, pp. 359–364.

Ravichandran, A., Chaudhry, R., Vidal, R., 2013. Categorizing dynamic textures using a bag of dynamical systems. Pattern Anal. Mach. Intell. 35 (2), 342–353.

Ravichandran, A., Favaro, P., Vidal, R., 2011. A unified approach to segmentation and categorization of dynamic textures. Asian Conf. Comput. Vision 6492, 425–438.

Saisan, P., Doretto, G., Wu, Y.N., Soatto, S., 2001. Dynamic texture recognition. Comput. Vision Pattern Recogn. 2, II–58–II–63.

Scherer, S., Rehder, J., Achar, S., Cover, H., Chambers, A., Nuske, S., Singh, S., 2012. River mapping from a flying robot: state estimation, river detection, and obstacle mapping. Autonom. Robots 33 (1-2), 189–214.

Schwind, R., 1991. Polarization vision in water insects and insects living on a moist substrate. J. Comp. Physiol. A 169 (5), 531–540.

Scott, D.W., 2009. Multivariate density estimation: theory, practice, and visualization, 383. John Wiley & Sons.

Sharan, L., Liu, C., Rosenholtz, R., Adelson, E., 2013. Recognizing materials using perceptually inspired features. Int. J. Comput. Vision 1–24.

Smith, A., Teal, M., Voles, P., 2003. The statistical characterization of the sea for the segmentation of maritime images. Video Image Process. Multimedia Commun. 2, 489–494.

Snoek, C., Worring, M., Smeulders, A., 2005. Early versus late fusion in semantic video analysis. In: International Conference on Multimedia. ACM, pp. 399–402.

Tenenbaum, J., de Silva, V., Langford, J., 2000. A global geometric framework for nonlinear dimensionality reduction. Science 290 (5500), 2319–2323.

Van Gemert, J.C., Veenman, C.J., Geusebroek, J.-M., 2009. Episode-constrained cross–validation in video concept retrieval. IEEE Trans. Multimedia 11 (4), 780–786.

Varma, M., Zisserman, A., 2005. A statistical approach to texture classification from single images. Int. J. Comput. Vision 62 (1), 61–81.

Varma, M., Zisserman, A., 2009. A statistical approach to material classification using image patch exemplars. Pattern Anal. Mach. Intell. 31 (11), 2032–2047.

Vidal, R., Ravichandran, A., 2005. Optical flow estimation and segmentation of multiple moving dynamic textures. Comput. Vision Pattern Recogn. 2, 516–521.

Zhao, G., Pietikäinen, M., 2007. Dynamic texture recognition using local binary patterns with an application to facial expressions. Pattern Anal. Mach. Intell. 29 (6), 915–928.

Zhao, G., Pietikäinen, M., 2007. Dynamic texture recognition using volume local binary patterns. In: Dynamical Vision. Springer, pp. 165–177.