Lecture Notes in Computer Science:
Local Trinary Patterns Algorithm for Moving Target Detection

Xuan Zhan and Xiang Li
Department of software engineering, East China Institute of Technology,
Nanchang, Jiangxi Province, China

Abstract. In this paper, we present a novel moving target detection called Local Trinary Patterns which is based on Local Binary Patterns algorithm. The standard LBP mainly captures the texture information, and in some circumstances it results in misidentification. The proposed LTP feature, in contrast, captures the gradient information and some texture information. Moreover, the proposed LTP are easy to implement and computationally efficient, which is desirable for real-time applications. Experiments show that this algorithm can significantly improve the detection performance and produce state of the art performance.

Keywords: moving target detection, local binary patterns, local trinary patterns, texture feature.

1 Introduction

The ability to detect moving target in images has a major impact on applications such as video surveillance, smart vehicles, robotics. Changing variations in moving target such as clothing, combined with varying cluttered backgrounds and environmental conditions, make this problem far from being solved.

The proposed method has a simple flow: every pixel at every frame is encoded as a short string of ternary digits (trits) by a process which compares this frame to the previous and to the next frame. The encoding process itself is based on comparing nearby patches, in a manner inspired by the self-similarity approach. For every pixel of every frame, a small patch centered at this pixel is compared to shifted patches in the previous and in the next frame. In a manner pertaining to the Local Binary Pattern approach, one trit of information is used to describe the relative similarity of the two patches to the patch in the central frame: the shifted patch in the previous frame is more similar to the central one, the patch in the next frame shifted by the same amount is more similar, or both are approximately comparable in their similarity.

2 The LBP Feature

LBP is a texture descriptor that codifies local primitives (such as curved edges, spots, flat areas) into a feature histogram. LBP and its extensions outperform existing texture descriptors both with respect to performance and to computational efficiency[1].
The standard version of the LBP feature of a pixel is formed by thresholding the $3 \times 3$ neighborhood of each pixel with the center pixel’s value. Let $g_c$ be the center pixel graylevel and $g_i$ (i=0,1,...,7) be the graylevel of each surrounding pixel. If $g_i$ is smaller than $g_c$, the binary result of the pixel is set to 0, otherwise to 1. All the results are combined to a 8-bit binary value. The decimal value of the binary is the LBP feature. See Fig. 1 for an illustration of computing the basic LBP feature.

![Fig. 1. Illustration of the basic LBP operator](image)

In order to be able to cope with textures at different scales, the original LBP has been extended to arbitrary circular neighborhoods by defining the neighborhood as a set of sampling points evenly spaced on a circle centered at a pixel to be labeled. It allows any radius and number of sampling points. Bilinear interpolation is used when a sampling point does not fall in the center of a pixel. Let $LBP_{p,r}$ denote the LBP feature of a pixel’s circular neighborhoods, where $r$ is the radius of the circle and $p$ is the number of sampling points on the circle. The $LBP_{p,r}$ can be computed as follows:

$$LBP_{p,r} = \sum_{i=0}^{p-1} S(g_i - g_c)2^i, S(x) = \begin{cases} 1 & \text{if } x \geq 0 \\ 0 & \text{otherwise.} \end{cases}$$

Here $g_c$ is the center pixel’s graylevel and $g_i$ (i=0,1,...,7) is the graylevel of each sampling pixel on the circle. See Fig. 2 for an illustration of computing the LBP feature of a pixel’s circular neighborhoods with $r = 1$ and $p = 8$. Ojala et al. proposed the concept of “uniform patterns” to reduce the number of possible LBP patterns while keeping its discrimination power. An LBP pattern is called uniform if the binary pattern contains at most two bitwise transitions from 0 to 1 or vice versa when the bit pattern is considered circular. For example, the bit pattern 11111111 (no transition), 00001100 (two transitions) are uniform whereas the pattern 01010000 (four