

# Video Analysis

## 1. Übungsaufgabe

Philipp Omenitsch, 1025659  
Marko Mlinaric, 0825603

### 1 Überblick zur Implementation und Source-Code

Die 1. Aufgabe bestand darin ein Verfahren zur Segmentierung von Vordergrundobjekten in Videostreams zu implementieren. Dafür haben wir zuerst den Versuch unternommen, die in der Angabe vorgestellte Methode *Color Mean and Variance* zu implementieren. Als wir allerdings auf Probleme gestoßen sind, haben wir als Alternative die *ViBe* Methode implementiert [1]. Ein Beispiel für unser Endresultat ist in Figure 1 zu sehen.

Wir haben C++ und OpenCV gewählt um die Aufgabe zu lösen. Unser Source-Code besteht aus den beiden Files `main.h` und `main.cpp`. Im Header-File `main.h` befinden sich Deklarationen von Methoden und globale Variablen, die wir verwendet haben. Der Großteil der Implementation befindet sich im Sourcefile `main.cpp`.

### 2 Color Mean and Variance

Bei der Color Mean and Variance Methode wird für jedes Pixel und jeden Farbkanal extra ein Modell gebaut. Jedes Pixel wird als eigenes Signal interpretiert, für das in der Initialisierungsphase eine Normalverteilung mit  $N(\mu, \sigma)$  gefunden wird.

**Adaptivität.** Um Veränderungen der externen Umstände in das Modell einfließen zu lassen, fließen Hintergrund- und Vordergrundpixel, verschieden stark gewichtet wieder in das Modell mit ein.

**Probleme.** Bei CMV hatten wir vor allem das Problem, dass wenn Pixel nicht als Vordergrund erkannt werden, sie als Hintergrundpixel entsprechend stark gewichtet werden und unser Modell an entsprechenden Stellen sehr schnell eine hohe Varianz adaptiert, die mehr oder weniger fast alles als Hintergrund detektiert.

### 3 ViBe

ViBe ist ein Sample-basierter Algorithmus zur Segmentierung von Vordergrundobjekten in Videostreams. Das Hintergrundmodell besteht aus  $width \times height \times nbSamples$

Samples, wobei *width* und *height* die Breite und Höhe des Videostreams bezeichnen und *nbSamples* ein wählbarer Parameter für die Anzahl an Samples pro Pixel ist.

**Initialisierung.** Für die Initialisierung des Hintergrundmodells reicht ein einzelnes Frame, da für jedes Pixel auch aus der näheren Umgebung gesampelt werden kann. In unserer Implementierung sampeln wir dennoch aus mehreren Frames, da durch die Natur der Angabe davon auszugehen ist, dass es immer eine gewisse Anzahl an Initialisierungsframes geben wird und sich das positiv auf unser Modell auswirkt.

Segmentiert wird, indem der Wert jedes Pixels mit den entsprechenden Samples verglichen wird. Es wird gezählt für wie viele Samples, die Distanz kleiner einem festgelegten *threshold* ist. Überschreiten diesen Wert mehr als `reqMatches` Pixel eines Samples, so wird angenommen, dass dieser Pixel zum Hintergrund gehört.

**Adaptivität.** Sollte das Pixel Teil des Hintergrund sein, so wird das Hintergrundmodell zufällig upgedatet. Im Schnitt jedes *n*-te Mal wird ein zufälliges bestehendes Sample durch den Wert des aktuellen Pixel ersetzt. Ebenso wird jedes *n*-te Mal ein zufälliges Sample eines zufälligen Pixels in der Nachbarschaft durch den Wert des aktuellen Pixel ersetzt. Letzteres führt dazu, dass der Hintergrund langsam in den Vordergrund überläuft. *n* wird im Code als `subsamplingFactor` bezeichnet. Siehe dazu auch Figure 3.

**Vergleich.** Wir haben durch diesen Algorithmus im Vergleich zu *Color Mean and Variance* subjektiv deutlich sichtbare Verbesserungen erfahren. Siehe dazu den direkten Vergleich mit CMV in Figure 2.

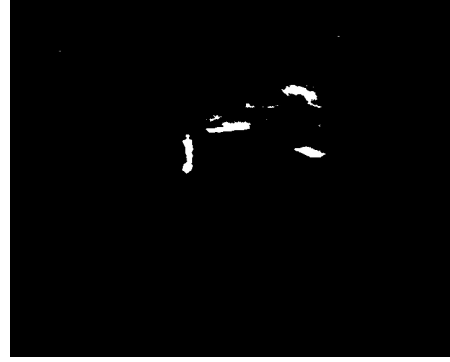
Um Rauschen noch etwas zu unterdrücken und Komponenten deutlicher darzustellen, wenden wir für jedes segmentierte Bild noch *Closing* an und füllen Konturen, um Löcher in den segmentierten Objekten zu schließen, da wir annehmen, dass die Objekte in der Szene keine großen Löcher haben.

## References

- [1] O. Barnich and M. Van Droogenbroeck, “Vibe: A universal background subtraction algorithm for video sequences,” *Image Processing, IEEE Transactions on*, vol. 20, no. 6, pp. 1709–1724, 2011.



(a) Originalbild

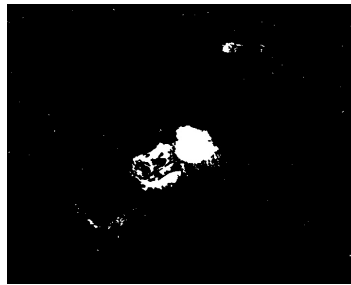


(b) Segmentierung

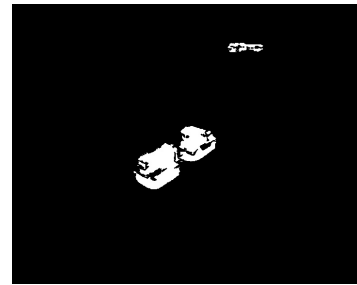
Figure 1: Vergleich Original und Segmentierung mittels ViBe.



(a) Originalbild



(b) CMV



(c) ViBe

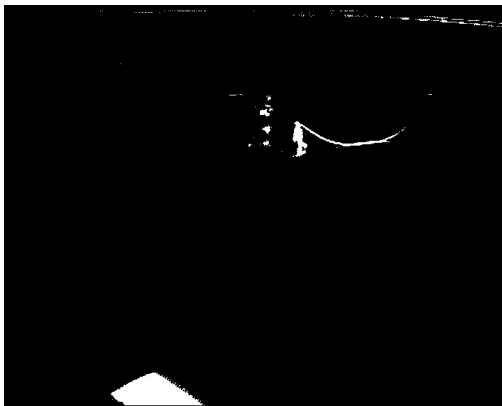
Figure 2: Vergleich Original, CMV und ViBe.



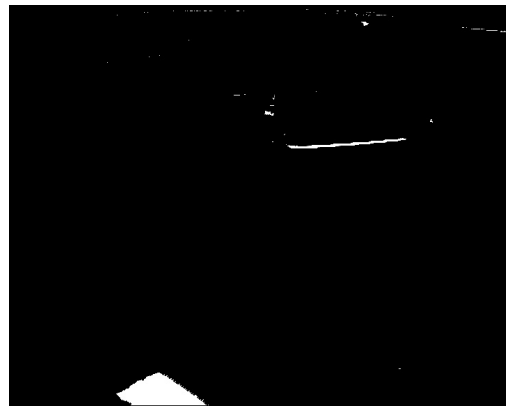
(a) Segmentierung für Frame #175



(b) Segmentierung für Frame #222



(c) Segmentierung für Frame #264



(d) Segmentierung für Frame #316



(e) Original Frame #175



(f) Original Frame #316

Figure 3: Adaption. Objekt verschwindet in den Hintergrund.