Object detection in motion-blurred images

Title:	Object detection in motion-blurred images
Student Full Name:	Darkhan Islam
Student ID:	11048993
Date:	25-January-2025
Copyright:	

Contents

List of Figures

1 Introduction

Motion blur is a common phenomenon that occurs when a moving object is captured with a relatively slow shutter speed, leading to a loss of sharpness and detail in the image. This effect is frequently encountered in various real-world scenarios, such as sports photography, surveillance systems, and autonomous driving applications. While motion blur can sometimes be used artistically to convey motion, it poses a significant challenge for object detection and tracking algorithms, which typically rely on sharp images for accurate processing.

Detecting and analyzing motion-blurred objects is an essential task in computer vision, with numerous practical applications. In autonomous systems, the ability to accurately detect moving objects despite motion blur is critical for making real-time decisions. Similarly, in video surveillance, motion blur can hinder the accurate identification and tracking of subjects. The challenge lies in effectively separating the blurred object from the background and estimating its motion parameters without prior knowledge of the object's trajectory.

A promising approach to tackling this challenge is inspired by the work of V. Caglioti and A. Giusti in their paper *"On the Apparent Transparency of a Motion Blurred Object"*, published in the *International Journal of Computer Vision* in 2010 [1]. The authors propose a mathematical model that interprets motion blur as an apparent semi-transparent layer, allowing for the decomposition of the blurred image into the background and moving object components. Their approach is based on the alpha matting model, which estimates the contribution of the moving object at each pixel location, enabling the extraction of its shape and trajectory.

In this work, we aim to implement and extend the method proposed by Caglioti and Giusti to detect moving objects in motion-blurred images. Rather than relying on conventional deblurring techniques, we leverage the alpha matting approach combined with geometric modeling to separate the blurred object from the background and recover its motion trajectory. To validate our method, we conduct experiments on both synthetic and real-world datasets, including a controlled simulation of object motion and the analysis of real-world images capturing the fall of a hairbrush.

The remainder of this report is structured as follows: Section 2 presents the problem formulation, including the alpha matting model and challenges in motion-blurred object detection. Section 3 describes the proposed solution approach, detailing the mathematical formulation and contour detection techniques. Section 4 covers the implementation aspects, including software tools, key processing steps, and performance optimization strategies. Section 5 discusses the experimental activities and results, including background extraction, segmentation, and trajectory estimation. Finally, Section 6 summarizes the findings and suggests directions for future work.

2 Problem Formulation

Motion blur is a prevalent issue in images captured in dynamic environments, where the relative movement between the camera and objects in the scene causes a streaking effect. This phenomenon significantly affects the accuracy of object detection and trajectory estimation, which are critical tasks in applications such as surveillance, autonomous driving, and robotic navigation. The blurred appearance of moving objects introduces ambiguity in localization and recognition, making it challenging to extract meaningful information from the images.

The process of image formation in the presence of motion blur can be mathematically modeled as a temporal integration of sharp image intensities over the camera exposure time. Given an image $C(x, y)$ captured by the camera, it can be expressed as:

$$C(x, y) = \int_{t_1}^{t_2} I(x_t, y_t, t) \cdot h(t)\, dt, \tag{1}$$

where:

- $I(x_t, y_t, t)$ represents the instantaneous sharp image of the scene at time t,
- $h(t)$ denotes the camera's exposure function, which determines how light is accumulated over time.

This formulation, which aligns with the temporal averaging model proposed in [1], highlights how motion blur results from the accumulation of multiple sharp instances of the object during the exposure period.

2.1 Alpha Matting Model

To address the problem of detecting blurred objects, an alternative approach is to consider the image as a composite of the moving object and the static background.

The goal of our approach is to estimate the alpha matte $\alpha(x, y)$, which encodes information about the object's shape and motion. This allows us to isolate the object from the background and reconstruct its trajectory even in the presence of severe motion blur.

2.2 Challenges in Motion-Blurred Object Detection

Detecting and tracking objects in motion-blurred images presents several challenges:

- **Loss of spatial details:** The streaking effect caused by motion results in the loss of fine object features, making traditional detection methods less effective.
- **Transparency effect:** Blurred objects appear semi-transparent due to the blending of background and object intensities, leading to difficulties in boundary estimation.
- **Unknown motion trajectories:** In real-world scenarios, the exact motion parameters (speed, direction, acceleration) are unknown, requiring robust estimation techniques.
- **Varying lighting conditions:** Differences in illumination can further complicate object-background separation.

2.3 Research Objectives

The primary objectives of this work are:

- To develop an effective method for detecting motion-blurred objects based on alpha matting techniques.
- To accurately estimate the trajectory of moving objects by analyzing their semi-transparent appearance in blurred images.
- To validate the proposed approach on both synthetic and real-world datasets, including controlled experiments such as tracking a falling hairbrush.

2.4 Scope of the Work

Our methodology is designed to work with both synthetic and real-world images, enabling thorough evaluation under controlled conditions as well as practical scenarios.

By formulating motion blur as a matting problem, we aim to provide a robust solution that can handle varying blur intensities, object speeds, and complex backgrounds, without relying on computationally expensive deblurring techniques.

3 Solution Approach

3.1 Overview of the Approach

Our methodology consists of the following key steps:

1. **Background Acquisition:** Instead of estimating the background from the blurred image, a separate photograph of the scene without the moving object was taken under the same lighting conditions.

2. **Foreground Separation:** The blurred image is processed to isolate the moving object using the alpha matting technique.

3. **Alpha Matte Estimation:** The transparency map $\alpha(x, y)$ is computed to represent the object's motion distribution.

4. **Contour Extraction and Tracking:** The object's contours are extracted from the alpha matte to determine its position over time.

5. **Trajectory Estimation:** The detected contours are used to fit a motion trajectory model.

3.2 Mathematical Formulation

In our approach, we represent the captured blurred image as a combination of the moving object and the static background using the alpha matting equation:

$$C(x, y) = \alpha(x, y) \cdot F(x, y) + (1 - \alpha(x, y)) \cdot B(x, y), \tag{2}$$

where:

- $C(x, y)$ is the captured blurred image,
- $B(x, y)$ is the separately captured background image,
- $F(x, y)$ is the unknown sharp object,
- $\alpha(x, y)$ is the unknown transparency map.

To estimate the alpha matte $\alpha(x, y)$, we apply the foreground approximation hypothesis, which assumes that the object intensity can be considered constant or known a priori:

$$\alpha(x, y) = \frac{C(x, y) - B(x, y)}{F(x, y) - B(x, y) + \epsilon}, \tag{3}$$

where ϵ is a small constant to prevent division by zero.

3.2.1 Hypotheses and Assumptions

Several key hypotheses are employed to solve for the unknowns in the alpha matting equation:

- **Foreground Approximation Hypothesis:** If the object moves slowly, its intensity can be considered constant over time. Mathematically, this can be expressed as:

$$\frac{\partial F(x, y, t)}{\partial t} \approx 0 \tag{4}$$

This allows us to approximate the foreground intensity as a fixed value.

- **Known Background Hypothesis:** Assuming the background remains unchanged during the motion, it can be expressed as:
$$B(x,y,t) = B(x,y) \tag{5}$$
This assumption allows us to directly subtract the background from the blurred image.
- **Contour Extraction Hypothesis:** We assume that object contours can be extracted using edge detection techniques by finding regions where the gradient of the alpha matte is significant:
$$\nabla\alpha(x,y) > \tau, \tag{6}$$
where τ is a predefined threshold value.

By applying these hypotheses, we successfully segment and track the moving object, ensuring accurate motion detection in motion-blurred scenarios.

3.3 Contour Detection and Motion Tracking

Once the alpha matte is estimated, contour detection techniques, such as Canny edge detection and morphological operations, are applied to extract the boundaries of the moving object. The extracted contours are then analyzed frame-by-frame to track the motion of the object.

The trajectory of the detected object is modeled using polynomial fitting or spline interpolation:
$$y = a_0 + a_1x + a_2x^2 + \ldots + a_nx^n, \tag{7}$$
where the coefficients a_i are determined by fitting the detected object positions over multiple frames.

4 Implementation

This section describes the technical details of the implementation of the proposed method for motion-blurred object detection and trajectory estimation. The implementation was carried out using Python, with extensive use of the OpenCV library for image processing and numerical computation libraries such as NumPy for mathematical operations.

4.1 Software and Tools

The implementation was developed using the following software tools and libraries:

- **Python 3.9:** The primary programming language used for the implementation.
- **OpenCV:** For image processing tasks such as filtering, contour detection, and morphological operations.
- **NumPy:** For efficient numerical computations and matrix operations.
- **Matplotlib:** For visualization and debugging of intermediate results.
- **Scikit-learn:** Used for trajectory fitting using polynomial regression techniques.

The development environment used for this project was Jupyter Notebook and VS Code, which allowed for interactive exploration and visualization of the data.

4.2 Implementation Workflow

The proposed method was implemented following the workflow outlined below:

1. **Image Acquisition:** A sequence of motion-blurred images was captured alongside a static background image under similar lighting conditions.
2. **Preprocessing:** The images were converted to grayscale, and noise reduction techniques were applied.
3. **Background Subtraction:** The captured background image was subtracted from each blurred image to isolate the moving object.
4. **Alpha Matte Estimation:** The alpha matting equation was applied to estimate the transparency map.
5. **Contour Detection:** The extracted alpha matte was processed to detect object contours.
6. **Trajectory Estimation:** The detected object contours were tracked across frames to estimate its trajectory.

4.3 Key Processing Steps

4.3.1 Preprocessing

The preprocessing pipeline consisted of the following steps to enhance the quality of the input images:

- **Grayscale Conversion:** All images were converted to grayscale to simplify processing and reduce computational complexity.
- **Gaussian Smoothing:** Applied to reduce high-frequency noise in both the blurred and background images.
- **Histogram Equalization:** Used to enhance contrast in low-light conditions.

4.4 Handling Shadow Noise

One of the major challenges encountered during the implementation was the presence of unwanted noise caused by shadows of the moving object, which were intensified by multiple light sources. These shadows appeared as duplicated contours in the background-subtracted image, which complicated the accurate estimation of the alpha matte and object boundaries.

To address this challenge, the following techniques were used:

- **Adaptive Thresholding:** Different regions were processed with variable thresholds to minimize shadow influence.
- **Contour Filtering:** Detected contours were filtered based on their shape and area to eliminate false positives caused by shadows.
- **Illumination Correction:** A preprocessing step involving local contrast normalization was applied to compensate for lighting variations.

4.5 Performance Optimization

Several optimizations were introduced to ensure real-time processing capabilities:

- Use of vectorized operations with NumPy to accelerate pixel-wise computations.
- Parallel processing for frame-wise operations using Python's multiprocessing module.
- Efficient memory management to handle high-resolution images without performance degradation.

4.6 Results Visualization

To facilitate debugging and analysis, the implementation included comprehensive visualization steps using Matplotlib:

- Visualization of intermediate processing steps such as background subtraction and alpha matte estimation.
- Overlaying detected contours on the original images to validate detection accuracy.
- Trajectory visualization using fitted polynomial curves.

4.7 Summary

The implementation of the proposed approach follows a structured pipeline that effectively isolates motion-blurred objects and estimates their trajectories. The use of a separately captured background image, combined with alpha matting and contour analysis techniques, allows for robust object detection even under challenging real-world conditions. Performance optimization techniques ensure the method can be deployed in real-time applications with minimal computational overhead.

5 Experimental Activity and Results

5.1 Overview

To evaluate the proposed motion blur detection and trajectory estimation approach, we conducted experiments on both synthetic and real-world data. The experimental setup involved the use of real-world photographs captured with a smartphone to simulate object motion and a synthetic background to aid in separating the foreground object.

5.2 Experimental Setup

The experiment consisted of capturing a falling comb using a smartphone camera. Additionally, a separate image of the background was taken to facilitate background subtraction. The challenging aspect of the captured images was the presence of multiple shadows caused by different light sources, adding complexity to the detection process.

Figure 1: Raw images of a falling comb captured by a smartphone. The comb experiences significant motion blur, while the shadows introduce noise.

5.3 Background Extraction

A separate image of the background was taken to assist in background subtraction. This background image contains noise in the form of shadows from hands and the comb, which creates challenges for the alpha matting approach.

Figure 2: Captured background image with noise from shadows.

5.4 Contour Detection

After subtracting the background from the raw images, contour detection was applied to isolate the blurred object. The detected contours are shown in the images below.

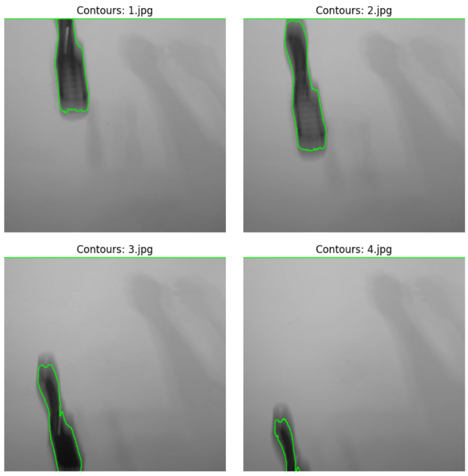

Figure 3: Detected contours of the comb across multiple frames. The green lines represent the object boundaries.

5.5 Binary Segmentation

To further refine object detection, grayscale thresholding was applied to extract the foreground object.

Figure 4: Binary segmentation results after background subtraction, isolating the comb from the scene.

5.6 Alpha Matting Process

The core of the approach involved applying the alpha matting method to extract the foreground object by estimating its transparency.

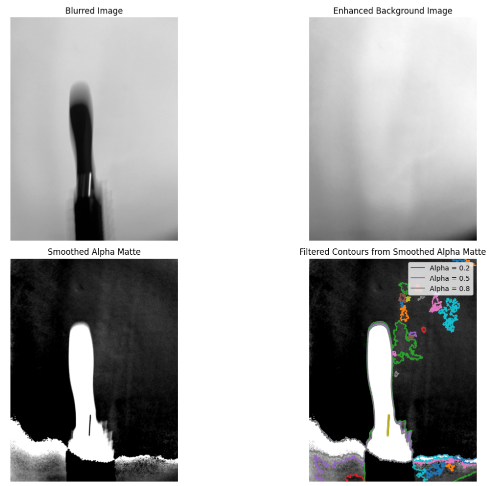

Figure 5: Alpha matting process showing the blurred image, enhanced background, smoothed alpha matte, and filtered contours.

5.7 Final Contour Refinement

Applying edge detection and thresholding techniques led to further refinement of the detected object contours.

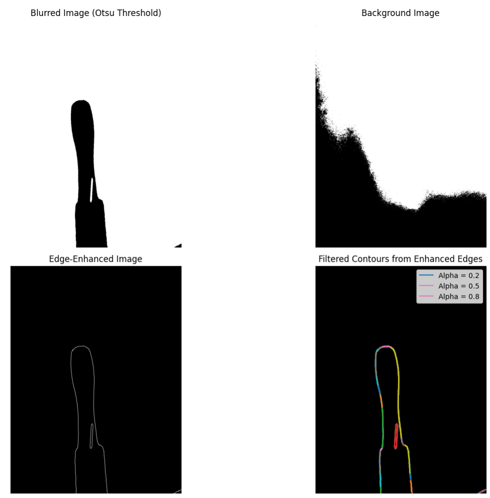

Figure 6: Final contour refinement process highlighting the object's shape using edge enhancement and Otsu thresholding.

5.8 Trajectory Estimation

Using the detected contours, we estimated the object's trajectory. The following figures present the detected and smoothed trajectory of the falling comb.

Figure 7: Detected object trajectory (blue) with smoothed trajectory (green) after processing the sequence of images.

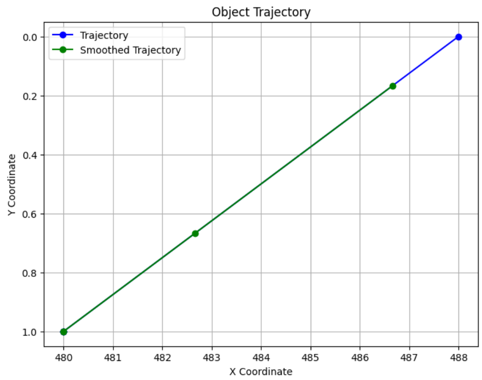

Figure 8: Final refined trajectory estimation showing the object's movement through the image sequence.

5.9 Discussion of Results

The proposed approach demonstrated robustness in detecting and tracking the comb despite the presence of noise from shadows. However, some limitations were observed, such as difficulty in distinguishing between the object and shadow regions in high-motion frames.

6 Conclusions and Future Work

This work presented an approach to detecting and analyzing motion-blurred objects using alpha matting and trajectory analysis.

Open Problems:

- Handling occlusions in multi-object scenarios.
- Extending the approach to 3D trajectory estimation using stereo images.
- Incorporating deep learning to improve alpha channel estimation.

References

[1] V. Caglioti and A. Giusti, "On the apparent transparency of a motion blurred object," *International Journal of Computer Vision*, vol. 86, pp. 243–255, 2010.

