

Probabilistic Inverse Dynamics for Blood Pattern Reconstruction

B. Cecchetto¹ and W. Heidrich¹

¹Department of Computer Science, University of British Columbia, Canada

Abstract

We present a method of reconstructing the region of origin and trajectories for particles given impact directions and positions. This method works for nonlinear trajectories, such as parabolic motion or motion with drag if given drag parameters. Our method works if given the impact speeds as well, or they can be estimated using a similar total initial energy prior. We apply our algorithm to the case of forensic blood pattern reconstruction, by automatically estimating impact velocities directly from the blood patterns. We validate our method in physically accurate simulated experiments, a feasibility study varying the impact angle and speed to estimate the impact speed from blood drop densities, as well as a forensic experiment using blood to reconstruct the region of origin.

Categories and Subject Descriptors (according to ACM CCS): I.5.4 [Pattern Recognition]: Applications—Computer Vision

1. Introduction

Consider a system of particles emitted from a common point, called a *region of origin*. The particles, without interfering with each other, undergo ballistic motion until they collide with a rigid object in the environment. A problem of interest is then to be able to reconstruct this region of origin, with only knowing the parameters of the particles as they impact surfaces. Such a problem is useful if we are interested in an event that took place in the past and could not observe it, but would like to know where it occurred based on observations we discern from the impact sites.

A prime example for this type of problem occurs in a branch of forensic science, called blood pattern analysis (BPA), which analyzes the patterns formed by blood particles eject from a wound. The impacts of blood droplets on surfaces leaves elliptical stains from which parameters such as direction and velocity of impact can be estimated. The reconstruction of the region of origin from these stains is called blood pattern reconstruction.

Traditional techniques for this problem involve identifying the direction of impact and extending that direction using physical strings or software. The region of origin is assumed to lie at the intersection of the flight paths. For this method to work, all particles must have a high enough velocity for the trajectories to approximate lines. For high veloc-

ity events such as exit spatter caused by a firearm, this works well, but for medium and low velocity events, droplets do not generally travel in straight lines and are influenced by both gravity and drag. It is generally known that the true region of origin will be lower than the intersection of these tangent lines. However, where exactly it may lie cannot be determined with the linear approach.

In this paper, we instead introduce a non-linear, probabilistic approach to solve the inverse dynamics problem posed by region of origin estimation. We begin by discussing previous work in Section 2, then examine the trajectory equation for a particle in Section 3. We then show how to use this equation to obtain a probability density function (PDF) over the space. We use the PDF from all impact sites to show the common likely region of the trajectories, which we assume to be the region of origin. In Section 4, we turn to the specific problem of BPA, and discuss how to obtain all the required parameters for our probabilistic inverse dynamics from photographs of blood patterns. In Section 5, we validate our method with experiments from real world data and test scenes.

2. Previous Work

In this section, we will discuss various related work in inverse dynamics in Section 2.1, traditional BPA methods in Section 2.2, and existing automated methods in Section 2.3.

2.1. Inverse Dynamics

Inverse dynamics is a problem with ongoing research mainly in robotics and locomotion. The joint torques of an articulated character are solved for, given a final state of the character in order to move the character from one pose to another as described in Featherstone's book [Fea87]. This problem differs from our setting in that we consider particles that are independent, and so we desire finding the forces and not torques. Other software such as BulletFlight [Arm] determines the ballistic path given a target. This software takes into account many different parameters such as humidity, temperature, bullet model, and others in order for a sniper to hit their target from large distances.

2.2. Traditional Blood Pattern Analysis

An introductory resource for BPA is Blood Dynamics by A. Wonder [Won01]. This book describes how to deduce accurate facts involving blood from a crime scene, as well as the basic linear reconstruction techniques. An updated book on the matter [BG08] covers more modern techniques including virtual stringing with software. Good supplements to this work are [EJ89] and [Won07], which focus more on case studies from various scenes.

Various BPA tools are presented in [BG08] using flowchart diagrams to deduce the order of events. Also presented are techniques for BPA reconstruction such as the tangent method which uses information from elliptical bloodstains. To perform this method, strings are run along the impact direction from each droplet's impact ellipse. One angle of the direction is given by the major axis angle in the plane it is on, while the other impact angle θ is determined by the ratio of the major axis length and width (L and W respectively) by the equation,

$$\sin(\theta) = \frac{W}{L}. \quad (1)$$

Another study related to traditional BPA is [Rog09] where the linear method was shown to be as accurate while varying white to red blood cell ratio (hematocrit values) in test stains.

2.3. Automatic Blood Pattern Analysis

The most commercial software in use is BackTrack [Car]. A user inputs bloodstain ellipses and positions in 3D. The program then computes a linear estimate of where the region of origin is. This software only works for axis aligned surfaces. Evaluations of this program have been done, comparing it to the stringing method [CFEH*06] and concluding that it is a reasonably accurate method for most results.

When gravity is a key factor in the trajectories, a different evaluation [ICLY05] concludes the height coordinate has to be approached with caution as an upper bound for the region of origin.

Shen [SBC06] describe an outline of various computer vision techniques as applied to BPA. They claim to obtain the region of origin, though no error results are provided for the fully reconstructed result. There is also a more recent paper [BKA10] describing an ellipse fitting technique specifically designed for blood droplet analysis. They obtain approximately 10% error in direction, testing on 30 stains. Another automatic approach fits homographies from coplanar ellipses from one image to another to infer information about the scene [WWRM06]. Their reconstructed results as applied to BPA were shown with errors of 35 and 67 inches. This is a significant error, as stated in [CFEH*06], they say 10-20cm (or 4-8 inches) is accurate enough to allow a proper interpretation of the crime.

There have been portable technologies to scan a crime scene more accurately as well, such as the DeltaSphere 3000 [Del], which obtains a dense model of the scene using a laser range scanner. It also captures color at the sampled points. There has also been a case study investigating the usefulness of virtual crime scene environments [MGH*98]. It concludes that creation of an interactive virtual environment of a crime scene to be very useful and of significant importance to many different related fields.

3. Region of Origin Reconstruction

In Section 3.1, we will discuss the trajectory equations, and how to use them to reconstruct the trajectory prior to an impact. Section 3.2 shows how to obtain a 2D PDF assuming we know all the parameters needed in the trajectory equations. We go on to show how to reconstruct the region of origin for the 2D case in Section 3.3. Lastly, we show in Section 3.4 how to obtain the 3D region of origin estimate given the 2D PDFs.

3.1. Inverse Dynamics

We assume a spherical particle that travels under the influence of gravity and drag only. From a physics textbook [HRK01], the trajectory for this motion is given by

$$\mathbf{c}_m(t) = \mathbf{v}_\infty t + \frac{m}{k}(\mathbf{v}_0 - \mathbf{v}_\infty)(1 - e^{-\frac{kt}{m}}) + \mathbf{x}_0. \quad (2)$$

where m is the mass of the particle, k is the drag coefficient, t is time, \mathbf{x}_0 and \mathbf{v}_0 is the position and velocity respectively at some time $t = 0$. We call this the main reconstruction path equation. The terminal velocity \mathbf{v}_∞ is given by

$$\mathbf{v}_\infty = \frac{m\mathbf{g}}{k}, \quad (3)$$

where \mathbf{g} is the gravity vector with final component $-g$, 0 elsewhere and $g = 9.81 \frac{m}{s^2}$ is the gravity constant.

We can fix this curve if we know the particle's mass m , drag coefficient k , impact velocity \mathbf{v}_0 and position \mathbf{x}_0 if we consider the time of impact to be $t = 0$. In this section we assume we know these parameters for each impact site. We show how to obtain them for blood specifically in Section 4, however this reconstruction method can work on any such system.

To reconstruct the trajectory prior to impact, we simply compute $\mathbf{c}_m(t)$ for $t < 0$. Note that the equation becomes unstable when the impact velocity is the terminal velocity. Intuitively, when a particle travels at terminal velocity it travels in a straight line. We cannot know how long it has been doing so as all the variables will be the same at the beginning and end of this line.

3.2. 2D PDF Formulation

If we reconstruct the region of origin for the reconstructed paths as in Section 3.1, we could find the least squared distance between all the curves. However, measurements may be noisy and a least squared estimate may greatly affect the reconstruction for the region of origin. In particular the reconstruction is very sensitive in reconstruction of the angle of impact. We choose a probabilistic formulation to encode the error in impact velocity, which is composed of the impact angle θ and speed v_0 . Let us assume we are in 2D space with x as the first and z as the second component of each vector. We write the velocity as

$$\mathbf{v}_0 = v_0(\cos(\theta), \sin(\theta)). \quad (4)$$

If we have a small positive or negative perturbation in the impact speed component δ_v and impact angle δ_θ , we can write this perturbed velocity as

$$\mathbf{v}_p(\delta_\theta, \delta_v) = (v_0 + \delta_v)(\cos(\theta + \delta_\theta), \sin(\theta + \delta_\theta)). \quad (5)$$

To reconstruct a perturbed path $\mathbf{c}_p(t)$ with this altered impact velocity \mathbf{v}_p , we can substitute it in for \mathbf{v}_0 in Equation 2.

If we want to know at the same point in time t how these two paths relate to each other, we can look at a perturbation vector for the paths $\mathbf{p}(t)$ which we define as

$$\mathbf{p}(t) = \mathbf{c}_p(t) - \mathbf{c}_m(t). \quad (6)$$

We can analyze what happens to this perturbation vector over time by substituting the path equations into Equation 6,

$$\mathbf{p}(t, \delta_\theta, \delta_v) = \frac{m}{k}(1 - e^{-\frac{kt}{m}})(\mathbf{v}_p(\delta_\theta, \delta_v) - \mathbf{v}_0). \quad (7)$$

The time dependent component $\frac{m}{k}(1 - e^{-\frac{kt}{m}})$ scales exponentially as we vary time in the negative direction. This means we are more uncertain of where the event occurred in space, the further in time before the event. The difference of the velocities is a circular warp from the space $\delta = (\delta_v, \delta_\theta)$

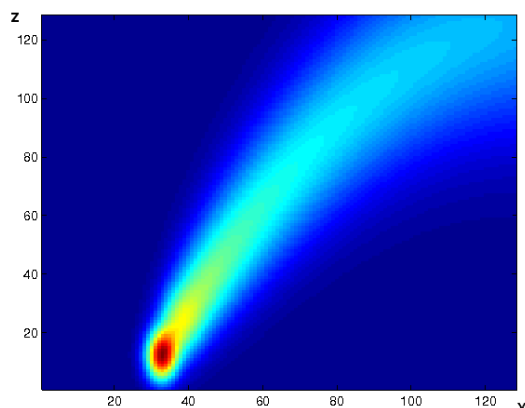


Figure 1: An example PDF for a particle over xz space.

to $\mathbf{x} = (x, z)$. Suppose the distribution of values in δ -space is a zero mean 2D Gaussian distribution with standard deviations σ_v and σ_θ in each component respectively, then we can map that distribution to xz coordinates space using $\mathbf{v}_p - \mathbf{v}_0$.

Since the perturbation values are relatively small, the warp is not pronounced and we can approximate the warped Gaussian in xz space by a uniform 2D Gaussian with mean $\mu_G(t)$ and standard deviation in each x and z axis as $\sigma_G(t)$. We know the mean of this distribution is where the perturbation is zero, which corresponds to the main reconstruction path $\mathbf{c}_m(t)$. We can thus say that $\mu_G(t) = \mathbf{c}_m(t)$. As for the standard deviation, we decide to choose to warp the positive and negative of the basis vectors in δ -space, and choose the one with maximal distance from the main reconstruction path

$$\sigma_G(t) = \max_{(p_\theta, p_v) \in \{-\sigma_\theta, \sigma_\theta\} \times \{-\sigma_v, \sigma_v\}} \|\mathbf{p}(t, p_\theta, p_v)\|. \quad (8)$$

This gets us a distribution for the probability of a point at a given point in time before the event. If we are interested in the probability of where a point has been over all times, we integrate over those times to obtain the PDF for a single particle,

$$P(x, z) = \int_{t \in [t_0 - t_E, t_0]} \mathcal{N}(\mu_G(t), \sigma_G(t)) dt, \quad (9)$$

where t_E is how far back in time we are interested in, and \mathcal{N} is the uniform 2D normal distribution with mean μ_G and standard deviation σ_G . This forms a PDF which denotes the probability of a particle being at a given point in space in some known time interval. An example of this PDF for a given impact site is seen in Figure 1. Note that we are more certain where it is closer to the impact site.

3.3. Region of Origin PDF Formulation

In the previous section we have created a probabilistic model of where a particle may have been for all times prior to impact in a given space. The highest probability is near the measured point. However, we wish to find the region of origin, or where it is likely to be for all points. We assume that all particles are independent observations from the same source, and therefore multiply all the values together for all particles.

Suppose we have a 2D probability density function for each impact particle denoted P_i for the i^{th} impact site. The input to these PDFs are known estimates for the position \mathbf{x}_0^i and velocity \mathbf{v}_0^i at the time of impact. To obtain the region of origin PDF, denoted P_{ROO} , we multiply all the P_i to find a common likely region all the particles will have been through,

$$P_{ROO} = \prod_i P_i. \quad (10)$$

We now show how to estimate the region of origin without knowledge of the velocities by imposing a similar initial total energy prior. We assume at the time of separation, similar kinetic energy, K , was added to all particles and since they also have the same potential energy, U , the total energy, T , for each particle is constant. To approximate the speed of a particle, s_i , given its mass m_i and height, h_i , relative to the lowest particle with height defined as 0 and speed s_0 , we can use the equation,

$$T = K + U = \frac{1}{2} m_0 s_0^2 + 0 = \frac{1}{2} m_i s_i^2 + m_i g h_i. \quad (11)$$

If we don't have knowledge of the masses, we can impose a similar mass constraint as well, obtaining

$$s_0^2 = s_i^2 + 2gh_i, \quad (12)$$

which can be solved for the speed of each particle s_i in terms of s_0 .

Suppose the approximate speed for the lowest particle is in an interval $S = [s_{start}, s_{end}]$. Then we construct a PDF by integrating over all different speeds,

$$P_v = \int_{s_0 \in S} P_{ROO}(s_0) ds. \quad (13)$$

The region of origin should lie where this new density function P_v is maximal, otherwise known as the maximum likelihood estimate (MLE).

3.4. 3D Region of Origin PDF

So far all of the PDFs have been for the 2D case. If we want to determine this probability distribution in 3D, we can multiply the values of the two perpendicular planes at different voxel values, then solve for the MLE in this 3D space. We define the PDF on the xz plane as

$$\pi_{xz} = P_v(\mathbf{x}_0^{xz}, \mathbf{v}_0^{xz}), \quad (14)$$

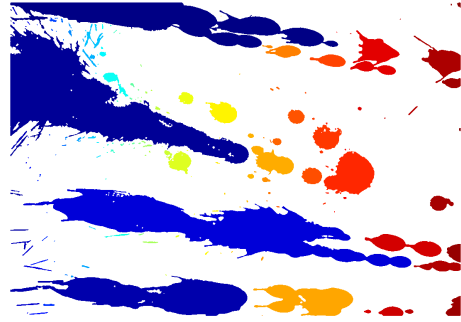


Figure 2: An example segmentation of the different connected bloodstains labeled by color

where \mathbf{x}_0^{xz} and \mathbf{v}_0^{xz} are the x and z components of the impact positions and velocities respectively. Similarly the yz PDF defined as

$$\pi_{yz} = P_v(\mathbf{x}_0^{yz}, \mathbf{v}_0^{yz}), \quad (15)$$

for the y and z components. Then we can write the 3D PDF as

$$P_{3D}(x, y, z) = \pi_{xz}(x, z) \pi_{yz}(y, z). \quad (16)$$

The assumed solution to the region of origin is most likely region that all particles have passed through at some point in time. This solution is defined as a maximum likelihood estimate solving for \mathbf{x}_A , and is written as

$$\mathbf{x}_A = (x_A, y_A, z_A) = \arg \max_{(x, y, z)} P_{3D}(x, y, z). \quad (17)$$

4. Bloodstain Parameter Estimation

In this section, we will describe how to obtain the parameters for the trajectory Equation 2 needed to compute the reconstruction. These were previously described in Section 3.1 as v_0 , θ , \mathbf{x}_0 , k and m . We show how to estimate an elliptical fit accurately by pruning a stain of the mask in Section 4.1. We then proceed to describe how to obtain the drag coefficient k and mass m of the blood droplet in Section 4.3. Finally we show how one may obtain the impact speed v_0 from photographs of the stains in Section 4.4.

4.1. Ellipse Pruning

Previous work has shown how to mask stains according to blood color [BKA10], so we will assume we are given a boolean mask of the bloodstains with value *true* if it is a stain and *false* otherwise. To process one stain at a time we perform connected components and examine each contiguous stain one at a time. The connected components for a mask can be seen in Figure 2.

Let us consider the mask M for one bloodstain. We perform a distance transform on this mask to find the region with largest inscribed circle (which is assumed to be inside the ellipse region). We now consider all pixels that are visible to this circle, the ones that are not visible get set to *false*, creating a visibility mask. This is performed by casting rays from points on the circle and the ones that leave the mask are not visible.

This utilizes a definition for convex regions, that all points in a convex region must be visible to all other points in the same region. The pixels in this visibility mask that correspond to edges in the original mask should be part of the main ellipse as they were visible to a large portion of the convex region. We then perform a standard ellipse fitting algorithm [FPF99] on the convex hull of these edge points. More appropriately, one could use an ellipse fitting algorithm specific to bloodstains such as in [BKA10], however the direct fitting performed remarkably well.

As discussed earlier, the ellipse gets us impact position \mathbf{x}_0 and impact angle θ .

4.2. Major Axis Direction Flipping

Although we have determined the ellipse fit for impact, the impact angle is still ambiguous in either direction along the major axis. From the trajectory equation 2, we know the particles must travel straight along the horizontal xy plane. We therefore project these two directions per stain onto the xy plane. We now have two sets of lines in the xy plane. The two sets of lines will converge in two distinct points. We can solve for two different points of intersection by clustering.

In Figure 3, one of the points will be in front of the wall, and another virtual one is behind as the two directions will be reflections about the plane normal in the xy plane. We will choose the point in front of the wall, and axis direction associated with that point for each ellipse. For a more general plane, the point that is closer to the plane will be the correct one.

Without resolving this ambiguity, the height component may diverge in the direction of the region of origin. This is more extreme in the case of particles with parabolic trajectory.

4.3. Mass and Drag Coefficient Estimation

With a confident ellipse, we can obtain the impact angle θ , \mathbf{x}_0 , k and m from it. We fit a stain with ellipse with with minor axis radius r . We can estimate the volume of the droplet from the equation for the volume of a sphere as

$$V = \frac{4}{3}\pi r^3. \quad (18)$$

We can use the volume to obtain the mass m of the droplet,

$$m = \rho_B V, \quad (19)$$

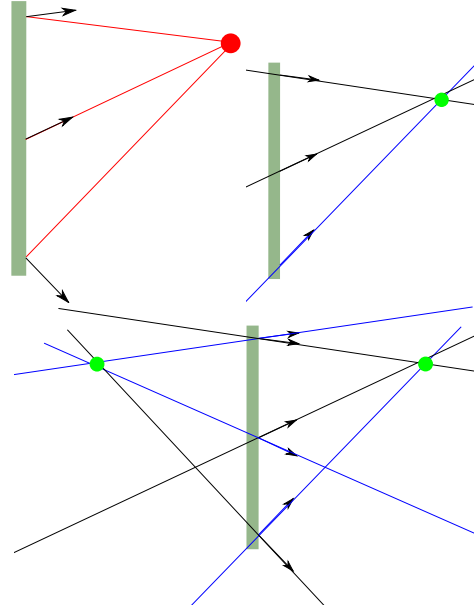


Figure 3: Axis flipping. The top left image is an example of input angles, the bottom image is considering all reflections of angles, the top right image corresponds to angles that meet on the correct side of the wall.

where ρ_B is the density of the given blood. Finally, the drag coefficient is defined as

$$k = 6\pi\mu r, \quad (20)$$

where μ is the dynamic viscosity for air ($1.78 \cdot 10^{-5} \frac{kg}{m \cdot s}$).

4.4. Estimating the Speed of Impact

If we would like to obtain the impact speed of a droplet, it is beneficial to define a density estimation for the droplet in image space, denoted ρ_i . This is formally defined as the thickness of a blood droplet at a given pixel. We can obtain this information from the Beer-Lambert law. This law states that for an incoming ray with intensity I_0 that passes through a medium with absorption coefficient α , and thickness l and outgoing ray I_1 , then

$$\frac{I_1}{I_0} = e^{-\alpha l}. \quad (21)$$

We assume the background is of uniform intensity. If we know the input intensity I_0 from the background and output intensity I_1 from the measured value on a blood droplet then, we can define the image density $\rho_i \equiv \frac{l}{2}$ as the light passes through the drop twice. Rearranging, we then obtain

$$\rho_i = -\frac{1}{2\alpha} \ln\left(\frac{I_1}{I_0}\right). \quad (22)$$

We claim the image density ρ_i is a function of both the

impact angle and impact speed. We know that there exists a maximum density of a droplet of a given size ρ_{max} , which corresponds to a perfect hemisphere on the surface. This occurs when the angle and speed of impact are both 0.

We know that as the speed increases, the density decreases as we're spreading the droplet over a larger area. As the speed approaches infinity, the density approaches zero. Similarly for the impact angle, as it approaches $\frac{\pi}{2}$, the stain will smear over an infinite distance and thus the density becomes zero. We can thus form ρ_i as follows that satisfy both these constraints,

$$\rho_i(\theta, v_0) = \rho_{max} \cos(\theta) e^{-Av_0}. \quad (23)$$

Since this is invertible, we can rearrange and solve for v_0 , and denote this as the velocity estimation equation,

$$v_0(\theta, \rho_i) = \frac{1}{A} \ln\left(\frac{\rho_{max} \cos(\theta)}{\rho_i}\right), \quad (24)$$

where the value A and ρ_{max} can be found through experiment (see next section and Figure 7).

5. Experimental Validation

In this section we show three experiments confirming our findings. Section 5.1 shows synthetic experiments gauging the accuracy of our region of origin estimation, Section 5.2 shows how our velocity estimation Equation 24 is a plausible model for data obtained by dropping blood at various speeds. Section 5.3 shows our full reconstruction technique for a horizontal scene.

5.1. Dynamics Simulations

We have tested the inverse dynamics aspect of our algorithm by simulating projectile motion for many particles from the same region of origin. The particles had random initial speeds and directions as well as termination time to simulate different impact positions in space. We reconstruct the the region of origin using our algorithm and compare with the traditional linear tangent method. For our method and the tangent method we run 100 trials per sample point and average the error. For our method without known velocities we run only 10 trials, as it takes much longer.

In Figure 4, we introduce error in the angle, σ_θ , and keep the error in the speed, σ_v , constant. In Figure 5 we introduce error in the speed. Note that it isn't appropriate to graph our method without speed in this case as we do not use it.

These trials may also be misleading in terms of usefulness. Although our algorithm becomes less accurate when increasing the error past a certain threshold, it also informs us how likely our estimate is, something the traditional method does not. Consider Figure 6. The PDF on the left is for one of the trials with low error in impact angles and velocities, whereas the PDF on the right is for one with high error in angle. The left one shows us it could have come from

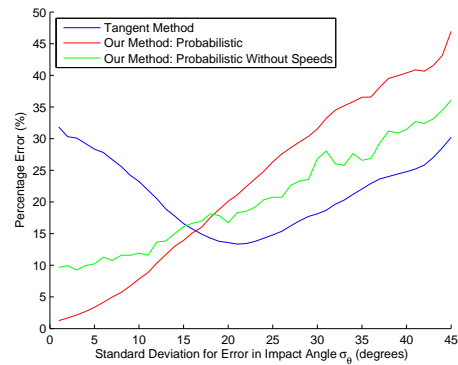


Figure 4: Synthetic experiment error - varying error in impact angle

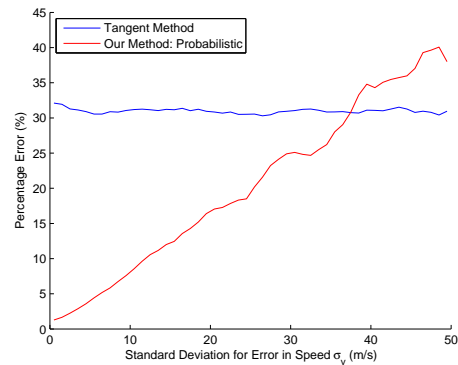


Figure 5: Synthetic experiment error - varying error in impact speed

one particular region whereas the one on the right shows many different regions, where the maximum likelihood estimate is not as useful.

5.2. Velocity Estimation Experiment

In this experiment, we dropped blood from an pipette at various heights to simulate various impact speeds. We did this for 5 different heights at 4 different angles. The results are shown in Figure 7. Cow blood was used in this and all of our other experiments.

If we examine the plot of density as a function of impact angle and speed, we can see that our model previously described fits the data appropriately. We compare it to the theoretical plot in Figure 8 for the same boundaries with value $\rho_{max} = 0.7598$ and $A = 0.1835$ with a residual norm of 0.0850.

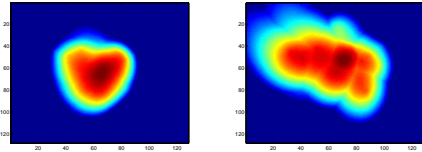


Figure 6: A PDF comparison between something with varying impact angle error $\sigma_\theta = 1deg$ and $\sigma_\theta = 45deg$

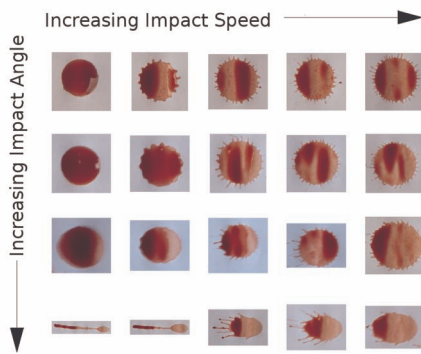


Figure 7: Photos of stains, varying the speed and angle of impact.

5.3. Reconstruction Experiment

The second experiment was designed to compare our region estimate against a ground truth. We hit a small puddle of cow blood with a hammer, varying the height of the puddle in each test. We ran our algorithm on the captured stain pattern.

In Table 1 we can see that the planar xy error for our method is comparable with the linear method, however we see a definite improvement in the height error z . Note that the z component error for the linear method may vary greatly depending on the configuration of the stains, whereas the probabilistic method remains at constant error.

Height	Error Type	Linear	Our Method
1.25	xyz	3.3768	1.1165
1.25	xy	0.5116	0.5789
1.25	z	3.3378	0.9547
5	xyz	4.9281	1.2866
5	xy	0.9619	1.2567
5	z	4.8333	0.2756
7.25	xyz	4.5954	2.3687
7.25	xy	2.2929	1.6903
7.25	z	3.9826	1.6594

Table 1: Errors for various heights for Euclidean error xyz , planar xy and height z . All units are in inches.

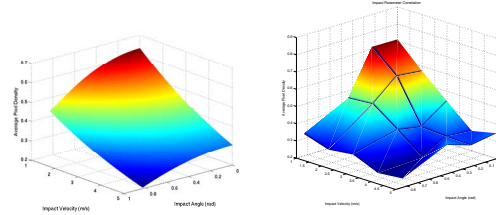


Figure 8: Comparison of theoretical (left) and experimental (right) data for varying the speed and angle of impact versus density of impact stains.

We also claim the error of our method is bounded by the accuracy of the xy error for the linear method. If this error is large, then it follows that we cannot perform an accurate reconstruction with the probabilistic method. This has been confirmed in other experiments where an xy estimate could not be obtained due to noise, and having a small test area relative to the region of origin position.

In Figure 9, we plot the region of origin and reconstructed flight paths. The paths in the 3d plots were estimated by parabolas. They were constructed using the estimated region of origin point, ellipse position and fixing the angle of impact, fixing all parameters of each parabola. This is why the linear method appears to be curved for some paths.

6. Conclusion and Future Work

6.1. Discussion and Limitations

We have described a method of reconstructing the region of origin for nonlinear trajectories, whereas no previous works have done so. It works with the same input data as in existing methods, the stain angles and positions in 3D space. We have also described a method to robustly calculate the impact angle given the image of a blood droplet, regardless of satellite stains. Furthermore, we have described a model to estimate blood droplet impact velocities.

The speed estimation is dependent on a given stain having all the blood from the associated particle contained in it. Essentially we have a volume constraint in the stain. This is not true, since some droplets may impact at an oblique grazing angle to the surface and rebound off further down in separate stains. It may also occur that the speed of a particle may be so high that it splashes into other nearby stains. If we do not use the speed estimation, we use a similar speed of impact prior, which is also a falsifiable assumption with extreme cases.

We also approximate a circular warp of a Gaussian with a uniform Gaussian for the particle probability equation 9 which may lead to inaccurate results.

Another limitation is as with many computer vision approaches, there are a few parameters to tune. For the re-

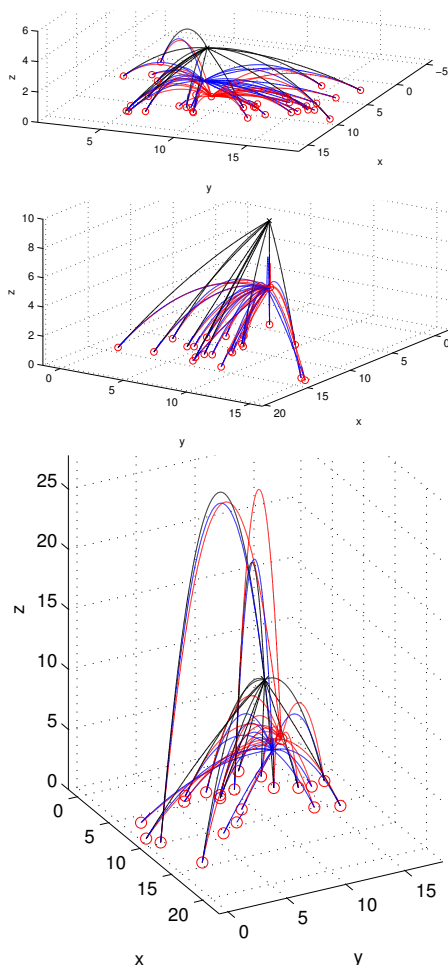


Figure 9: Reconstruction results using the linear method in black, our method in blue and the ground truth in red.

construction, we have to tune the δ_θ and δ_v values. For the stain segmentation, manual thresholds were used to mask the stains, even on a white background.

6.2. Future Work

More controlled tests are needed to accurately estimate impact speed based on stain density and angle, as described in Section 4.4. Also, more thorough validation of the whole method would also be necessary before using the presented algorithm in the field. In addition to accurately estimating the velocities, we could also learn δ_θ and δ_v which are the standard deviation in the angle and speed estimates by comparing it with ground truth data.

Right now, we have only tested on planar data to ease with calibration, although with a more rigorous capture setup it could be possible to acquire a 3d textured model of a scene.

To estimate the blood droplet angles from arbitrary surfaces would be very beneficial. Also, we have assumed one splatter event coming from a point source. If there were multiple events in the scene, or a blood trail, a new approach would be needed.

References

- [Arm] ARMAMENT K.: Bullet flight. <http://www.knightarmco.com/bulletflight/>. 2
- [BG08] BEVEL T., GARDNER R.: *Bloodstain pattern analysis: with an introduction to crime scene reconstruction*. CRC Press, 2008. 2
- [BKA10] BOONKHONG K., KARNJANADECHA M., AIYARAK P.: Impact angle analysis of bloodstains using a simple image processing technique. *Songklanakar Journal of Science and Technology* 32 (2010). 2, 4, 5
- [Car] CARTER A.: Backtrack images. <http://www.physics.carleton.ca/carter/>. 2
- [CFEH*06] CARTER A., FORSYTHE-ERMAN J., HAWKES V., ILLES M., LATURNUS P., LEFEBVRE G., STEWART C., YAMASHITA B.: Validation of the BackTrack Suite of Programs for Bloodstain Pattern Analysis. *Journal of Forensic Identification* 56, 2 (2006), 242. 2
- [Del] DELTASPHERE: Deltasphere 3000 - 3d scene digitizer. <http://www.deltasphere.com/DeltaSphere-3000.htm>. 2
- [EJ89] ECKERT W., JAMES S.: *Interpretation of bloodstain evidence at crime scenes*. Elsevier Science Ltd, 1989. 2
- [Fea87] FEATHERSTONE R.: *Robot dynamics algorithms*. Kluwer Academic Publishers, 1987. 2
- [FPF99] FITZGIBBON A., PILU M., FISHER R.: Direct least square fitting of ellipses. *IEEE Transactions on pattern analysis and machine intelligence* 21, 5 (1999), 476–480. 5
- [HRK01] HALLIDAY D., RESNICK R., KRANE K. S.: *Physics*. Wiley, 5th edition, 2001. 2
- [ICLY05] ILLES M., CARTER A., LATURNUS P., YAMASHITA A.: Use of the Backtrack Computer Program for Bloodstain Pattern Analysis of Stains From Downward-Moving Drops. *Canadian Society of Forensic Science Journal* 38, 4 (2005), 6. 2
- [MGH*98] MURTA A., GIBSON S., HOWARD T., HUBBOLD R., WEST A.: Modelling and rendering for scene of crime reconstruction: A case study. In *Proceedings Eurographics UK* (1998), pp. 169–173. 2
- [Rog09] ROGERS N.: Hematocrit—Implications for Bloodstain Pattern Analysis. 2
- [SBC06] SHEN A., BROSTOW G., CIPOLLA R.: Toward Automatic Blood Spatter Analysis in Crime Scenes. In *Crime and Security, 2006. The Institution of Engineering and Technology Conference on* (2006), pp. 378–383. 2
- [Won01] WONDER A.: *Blood dynamics*. Academic Pr, 2001. 2
- [Won07] WONDER A.: *Bloodstain pattern evidence: objective approaches and case applications*. Elsevier/Academic Press, 2007. 2
- [WWRM06] WRIGHT J., WAGNER A., RAO S., MA Y.: Homography from coplanar ellipses with application to forensic blood splatter reconstruction. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (2006), vol. 1. 2