Friction Prediction via Illumination Ratios

Shayan Nasiri
University of Toronto
s.nasiri@mail.utoronto.ca

Abstract—We study predicting surface friction from a single RGB image, a task important for safe robot locomotion and manipulation. Existing vision-based methods either rely on additional hardware, or depend on material recognition and word embeddings, which limits real-time deployment. In this project we revisit Brandão et al.'s GTF dataset and ask whether simple, illumination-aware image statistics can directly predict coefficient of friction (CoF). Our approach is based on the observation that apparent reflectance and micro-texture correlate with slipperiness, while raw intensity is strongly affected by shading. We simulate colored point-light illumination using a von Kries diagonal transform and compute luminance images for white, red, green, and blue illuminations. Taking log luminance ratios between these images suppresses shading and results in an illumination-invariant feature. We then extract the mean or standard deviation of the log-ratio in the region of interest and train a linear regression model to predict CoF. On the GTF dataset, the $log(Y_r/Y_w)$ illumination ratio achieves an average RMSE of 0.163, improving over Brandão et al.'s shading and gradient baselines (0.172 to 0.177) and slightly outperforming a DINOv2 feature baseline (0.165), while being far simpler and cheaper to compute. Adding the ratio to DINOv2 features provides no further gain, suggesting that such illumination cues are already included in modern deep features.

Index Terms—friction prediction, coefficient of friction, illumination ratios, von Kries transform, computer vision, DINOv2, robotics

1 Introduction

FRICTION estimation is an essential factor in the safe operation of legged robots and autonomous vehicles. Suppose the coefficient of friction (CoF) of a surface that a robot will come into contact with is overestimated or underestimated. In that case, the robot may either move too conservatively or slip and fall. Most deployed methods for estimating surface slipperiness use force/torque or tactile sensing after contact has already been made, which does not allow safe route planning in advance. As a result, a complementary goal is to predict surface friction from vision alone before contact, using only lightweight cameras that can be mounted on mobile robots.

Recent work has shown that visual cues are indeed informative about slipperiness. Brandão et al. [1] introduced the ground-truth coefficient of friction (GTF) dataset. They proposed simple image features derived from a Retinex shading layer (ShadMax, ShadStd, and GradMu) as baselines for CoF prediction, as well as a stronger method that combines material labels with word embeddings of material names. Although effective, their best-performing approach assumes that a semantic material label is known or can be guessed, which is difficult to guarantee in online robotic settings. In another approach, Zhang et al. [2] use a custom optical setup to capture reflectance disks that show how a surface reflects illumination from many directions, and train a CNN on these disks to predict CoF. This approach results in accurate predictions, but relies on bulky hardware that is not easily integrated into compact robotic platforms.

A separate line of work on color constancy and intrinsic image decomposition studies how changes in illumination affect image intensities. Finlayson et al. [3] showed that many illumination changes can be approximated by a simple channel-wise diagonal or von Kries transform applied in RGB space. At the same time, modern self-supervised vision models such as DINOv2 [4] learn powerful generic image representations that capture texture, reflectance, and geometry from large-scale image collections, and provide a strong baseline for many vision tasks without task-specific feature design.

In this project, we explore whether simple cross-illumination luminance ratios, derived from von Kries transforms, can improve CoF prediction on the GTF dataset while remaining lightweight enough for real-time deployment. Starting from Brandão et al.'s shading-based baselines, we linearize the input sRGB image, simulate multiple colored illuminants via per-channel gains, and compute luminance images under white, red, green, and blue virtual lights. We then form log-ratios such as Y_r/Y_w that are designed to suppress shading while emphasizing differences in apparent reflectance. Simple summary statistics (mean or standard deviation over the region of interest) of these ratio images are used as features to train a linear regression model for CoF prediction. To contextualize our hand-crafted features against modern deep representations, we also evaluate DINOv2 features extracted from the cropped regions and a hybrid model that augments DINOv2 with our best-performing illumination ratio feature.

Our main technical contributions are: (i) a systematic adaptation and evaluation of illumination-ratio features for friction prediction on the GTF dataset, including design choices such as linearization, von Kries gains, log-ratio formulation, and luminance masking; (ii) an empirical comparison against the original shading baselines of Brandão et al. and a strong DINOv2 regression baseline; and (iii) an analysis of residuals and outliers that highlights where ratio-based cues are most and least informative. Experiments show that a single mean feature derived from the red over white luminance ratio modestly but consistently outperforms all shading baselines and even slightly outperforms DINOv2, while being computationally simple and requiring

only a single RGB image. At the same time, our study is limited by the static and controlled nature of the GTF dataset, the use of linear regression, and the assumption of a known region of interest. We discuss these limitations and potential extensions to more realistic robotic settings in the conclusion.

2 Related Work

2.1 Vision Based Friction Estimation

Early work on predicting slipperiness from vision is by Brandão et al., who introduced the Ground-truth coefficient of Friction dataset (GTF) and systematically evaluated algorithmic and human performance on friction estimation from images [1]. As shown in Figure 1, GTF contains a collection of images from surface patches with different materials and viewpoints. For each image, the dataset includes a copy of the image with the region of interest (ROI) indicated by a square, as well as the cropped version of the ROI. Lastly, the dataset contains a .mat file that stores the accurately measured coefficients of friction and coordinates for the ROI in each image. Brandão et al. proposed several simple, hand-crafted image features computed over each ROI: ShadMax and ShadStd, which are the maximum or standard deviation of pixel values extracted from a Retinex-based shading layer, and GradMu, which is the mean of gradient magnitudes in the ROI computed using a Sobel filter [1]. These features form the baselines that we also implement and compare against in our experiments.

In addition to shading and gradient cues, Brandão et al. explored the use of semantic material labels and word embeddings. Their WordMM and WordMS methods first identify the material category of a patch (wood, tile, metal, etc.), then use distances in a word-embedding space between the query material's name and material with known CoF to predict friction [1]. While these methods achieve the best performance reported in their work, they require either ground-truth material labels or a material recognition system. This dependency makes them challenging to deploy in online robotic settings where material labels are not available, and visual appearance may be ambiguous.

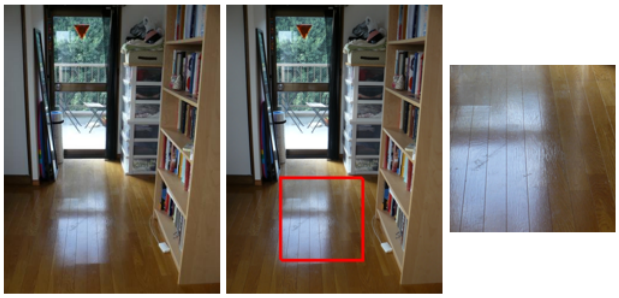

Fig. 1. Example image from the GTF dataset: (left) full scene, (middle) same image with the ground-truth region of interest (ROI) used for friction prediction highlighted in red, and (right) cropped ROI patch on which all features are computed.

Zhang et al. took a different approach and directly measured rich reflectance information to predict friction from a single capture [2]. As shown in Figure 2, they designed an optical system consisting of a camera, a beam splitter, and a parabolic mirror to acquire reflectance disks, in which each pixel corresponds to the reflected intensity for a particular incident light direction. These disks are then fed to a convolutional neural network that learns a mapping from reflectance to CoF [2]. The method achieves strong performance and demonstrates that detailed reflectance cues are highly informative about CoF. However, the required hardware is bulky, carefully calibrated, and constrained to a fixed imaging geometry, which makes it unsuitable for lightweight, mobile robotic platforms.

Our work is closest in spirit to Brandão et al. [1], in that we operate on the GTF dataset and aim to extract simple features from a single RGB image. Unlike [1], we do not assume access to material labels or external semantic knowledge to improve CoF prediction beyond the introduced shading-based baseline. Compared to the reflectance-disk approach of Zhang et al. [2], we aim to improve CoF prediction without relying on additional bulky hardware so that the method can be realistically deployed on robots.

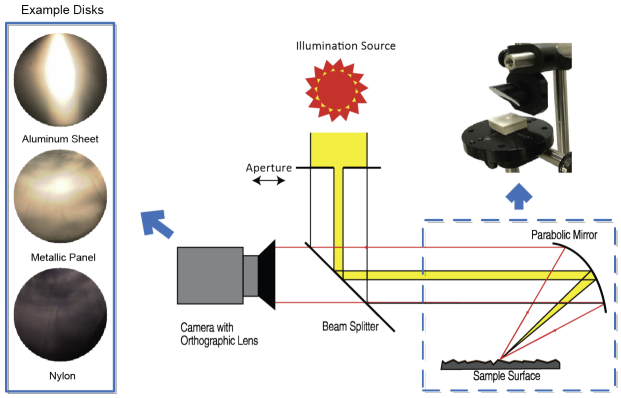

Fig. 2. Zhang et al.'s setup for capturing reflectance disks for different surface materials

2.2 Illumination-Invariant Reflectance and Color Ratios

Several works have explored capturing scenes under multiple illumination conditions to disentangle shading, shape, and reflectance. For instance, Cao et al. [5] proposed a method that combines stereoscopic vision with flash photography. By capturing a no-flash stereo pair to estimate coarse geometry and a corresponding flash/no-flash image pair to refine surface normals, their approach recovers both shape and albedo with high fidelity. Such multi-illumination strategies demonstrate that explicitly varying lighting provides strong constraints for separating intrinsic surface properties from shading artifacts.

In spirit, our approach shares the goal of exploiting multiple illuminations to reveal reflectance structure, but we aim for a significantly simpler and more practical implementation. Instead of physically varying the lighting or requiring active hardware like flashes, we approximate the effect of distinct colored illuminants using von Kries transforms applied to a single RGB image [3]. This allows us to retain the computational and hardware simplicity of Brandão et al.'s single-image setting [1], while borrowing

the intuition from methods like Cao et al. [5] that illumination variation is a powerful cue for isolating material properties.

3 Proposed Method

3.1 Problem Setup and Dataset

We follow the problem setup of Brandão et al. [1] and work on the GTF dataset. Each example consists of a color image of a surface patch, the pixel coordinates of a square ROI that indicates where a robot foot would contact, and a measured static CoF $y \in \mathbb{R}$ for that patch. In addition, the dataset provides cropped versions of the ROIs, which we use throughout our experiments.

Formally, for each image we observe a cropped RGB patch

$$I \in [0,1]^{H \times W \times 3}, \tag{1}$$

obtained by converting the 8-bit ROI image to the range $[0,1]$, and a scalar label y. The goal is to learn a regression function

$$f(I) \approx \tilde{y} \tag{2}$$

that predicts CoF from a single RGB image. We restrict ourselves to lightweight feature extractors and simple linear regression models so that the method remains suitable for real-time deployment on a mobile robot.

3.2 Image Pre-processing

3.2.1 sRGB Linearization

The GTF images are stored in sRGB, which has a nonlinear gamma curve applied. Before computing physical quantities such as luminance, we convert each image I to a linear RGB image $\tilde{I}$ using the standard sRGB inverse transfer function

$$\tilde{I}(p) = \begin{cases} \frac{I(p)}{12.92}, & I(p) \le 0.04045, \\ \left(\frac{I(p)+0.055}{1.055}\right)^{2.4}, & I(p) > 0.04045, \end{cases} \tag{3}$$

applied independently to each color channel and pixel p.

3.2.2 Luminance

From the linear RGB image we compute a luminance image Y_w (corresponding to a white illuminant) using the Rec. 709 coefficients

$$Y_w = 0.2126\,\tilde{I}_R + 0.7152\,\tilde{I}_G + 0.0722\,\tilde{I}_B \tag{4}$$

where $\tilde{I}_R, \tilde{I}_G, \tilde{I}_B$ are the red, green, and blue channels of $\tilde{I}$ respectively.

3.3 Shading-based Baseline Features

Brandão et al. [1] proposed three simple features computed over the ROIs: two based on a shading layer computed using the Retinex algorithm (ShadMax and ShadStd) and one based on gradient magnitude (GradMu) computed using a Sobel filter. We reimplement these baselines to compare against our illumination ratio features.

3.3.1 Shading Estimate

We estimate a log-shading image by smoothing the luminance and applying a logarithm,

$$S = \log\left(G_\sigma * Y_w + \varepsilon\right), \tag{5}$$

where G_σ denotes a Gaussian filter with standard deviation σ, $*$ is convolution, and ε is a small constant to avoid numerical issues. Lastly, the ROI coordinates from the GTF annotations are used to crop S to the contact region.

From the cropped shading image S_{roi} we compute

$$\text{ShadMax} = \max_{p \in \text{roi}} S_{\text{roi}}(p), \tag{6}$$

and

$$\text{ShadStd} = \operatorname*{std}_{p \in \text{roi}} S_{\text{roi}}(p) \tag{7}$$

3.3.2 Gradient Magnitude

For GradMu we compute the Sobel gradient of the luminance image Y_w inside the ROI,

$$\nabla Y_w = (g_x, g_y), \qquad M(p) = \sqrt{g_x(p)^2 + g_y(p)^2} \tag{8}$$

and take the mean gradient magnitude,

$$\text{GradMu} = \operatorname*{mean}_{p \in \text{roi}} M(p). \tag{9}$$

The three scalar features ShadMax, ShadStd, and GradMu, used individually in [1], form our shading-based baselines.

3.4 Cross Illumination Ratio Features

As shown in Figure 3, our main contribution is a family of features based on luminance ratios across virtual colored illuminants. The design is motivated by diagonal color-constancy models: Finlayson et al. [3] showed that many changes in illumination color can be modeled by a channel-wise von Kries transform in RGB space. We use such transforms to simulate multiple colored lights from a single input image, then form log ratios of luminance that are less sensitive to overall shading and more sensitive to apparent reflectance.

3.4.1 Simulating Illuminant Changes

Given the linear RGB image $\tilde{I}$ computed in equation 3, we define three sets of per-channel gains:

$$g_R = (1.6, 0.6, 0.6), \tag{10}$$

$$g_G = (0.6, 1.6, 0.6), \tag{11}$$

$$g_B = (0.6, 0.6, 1.6). \tag{12}$$

Applying a von Kries transform with each of the gains g_R, g_G, and g_B, produces three new linear RGB images

$$\tilde{I}_R = \text{clip}\left(g_R \tilde{I}, 0, 1\right) \tag{13}$$

$$\tilde{I}_G = \text{clip}\left(g_G \tilde{I}, 0, 1\right) \tag{14}$$

$$\tilde{I}_B = \text{clip}\left(g_B \tilde{I}, 0, 1\right) \tag{15}$$

with clip being a clipping operator that keeps values within the range $[0, 1]$. Intuitively, g_R approximates a reddish illuminant, g_G a greenish illuminant, and g_B a bluish illuminant.

From these images we compute the respective luminance:

$$Y_r = \mathcal{L}(\tilde{I}_R), \tag{16}$$
$$Y_g = \mathcal{L}(\tilde{I}_G), \tag{17}$$
$$Y_b = \mathcal{L}(\tilde{I}_B), \tag{18}$$

where $\mathcal{L}(\cdot)$ denotes the linear luminance operator introduced in equation 4.

3.4.2 Log-Ratio Images

Let Y_w be the luminance under the original (white) illuminant and Y_r, Y_g, Y_b be the luminances under simulated red, green, and blue lights. For each pixel we form log-ratio images

$$R_{rw}(p) = \log\big(Y_r(p) + \varepsilon\big) - \log\big(Y_w(p) + \varepsilon\big), \tag{19}$$
$$R_{gw}(p) = \log\big(Y_g(p) + \varepsilon\big) - \log\big(Y_w(p) + \varepsilon\big), \tag{20}$$
$$R_{bw}(p) = \log\big(Y_b(p) + \varepsilon\big) - \log\big(Y_w(p) + \varepsilon\big), \tag{21}$$

where ε is a small constant. If both Y_r and Y_w are scaled by the same multiplicative shading factor at a pixel, this factor cancels in the difference, so these log-ratios are designed to suppress shading while retaining differences in apparent reflectance across wavelengths.

Very dark or saturated pixels are unreliable for ratio computation. We therefore construct a simple luminance mask

$$\mathcal{M} = \{p \mid 0.02 < Y_w(p) < 0.98\} \tag{22}$$

and only use pixels in $\mathcal{M}$ when computing features.

3.4.3 Ratio Statistics as Features

From each masked log-ratio image we compute summary statistics over the ROI. Given a log-ratio map $R(p)$, we define

$$\mu_R = \operatorname*{mean}_{p \in \mathcal{M} \cap \text{roi}} R(p), \tag{23}$$
$$\sigma_R = \operatorname*{std}_{p \in \mathcal{M} \cap \text{roi}} R(p). \tag{24}$$

In principle, any combination of $\{\mu_{R_{rw}}, \mu_{R_{gw}}, \mu_{R_{bw}}\}$ or $\{\sigma_{R_{rw}}, \sigma_{R_{gw}}, \sigma_{R_{bw}}\}$ can be used as a feature vector. In our experiments, we systematically evaluate all possible subsets (all three, any two pairs, or a single ratio) and both mean and standard deviation statistics. We ultimately find that a single one dimensional feature (the mean of the red over white log-ratio $\mu_{R_{rw}}$) results in the best and most stable performance, while remaining extremely cheap to compute.

3.5 Regression Models

3.5.1 Linear Regression for Handcrafted Features

For the shading features and illumination-ratio features, we use a simple linear regression model. Let $\mathbf{x}_i \in \mathbb{R}^d$ be the feature vector for image i, and let y_i be the corresponding CoF. We fit

$$\hat{y}_i = \mathbf{w}^\top \mathbf{x}_i + b \tag{25}$$

by ordinary least squares on the training folds, using scikit-learn's `LinearRegression`. This choice matches the baseline in [1] and isolates the contribution of the feature representation rather than the regressor.

3.5.2 DINOv2 Deep Feature Baseline

To contextualize our handcrafted features against modern learned representations, we also evaluate a regression model built on top of DINOv2 features. Using the ViT-B/14 DINOv2 model, we extract a global pooled embedding $\mathbf{z}_i \in \mathbb{R}^D$ from each cropped ROI image, keeping the network frozen. We then train a ridge-regression model

$$\hat{y}_i = \mathbf{w}^\top \mathbf{z}_i + b \tag{26}$$

with an ℓ_2 penalty on $\mathbf{w}$, selecting the regularization coefficient by cross-validation within each training fold. Finally, we consider a hybrid model that concatenates our best illumination-ratio feature with the DINOv2 embedding, i.e.,

$$\tilde{\mathbf{z}}_i = [\,\mathbf{z}_i, \mu_{R_{rw},i}\,], \tag{27}$$

to test whether the ratio carries information beyond what is captured by DINOv2.

The evaluation protocol, 10-fold cross-validation with RMSE metric, and quantitative comparisons between the different models are presented in the next section.

4 Experimental Results

4.1 Evaluation Protocol

All experiments are conducted on the GTF dataset introduced by Brandão et al. [1]. For each image, we use the cropped ROI patch and the corresponding measured CoF as the regression target. Following [1], performance is reported in terms of root mean squared error (RMSE) between predicted and ground-truth CoF:

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\big(\hat{y}_i - y_i\big)^2} \tag{28}$$

where y_i and $\hat{y}_i$ denote the ground-truth and predicted CoF for sample i.

We employ 10-fold cross-validation with random shuffling: the dataset is partitioned into 10 folds, and in each run 9 folds are used for training and 1 fold for testing. The same folds are used for all methods to ensure a fair comparison. For all handcrafted features (shading and illumination ratios) we fit a linear regression model using scikit-learn's `LinearRegression`. For DINOv2 features we use ridge regression with the regularization parameter selected from a logarithmic grid of values between 10^{-3} and 10^{3} via cross-validation on the training folds only.

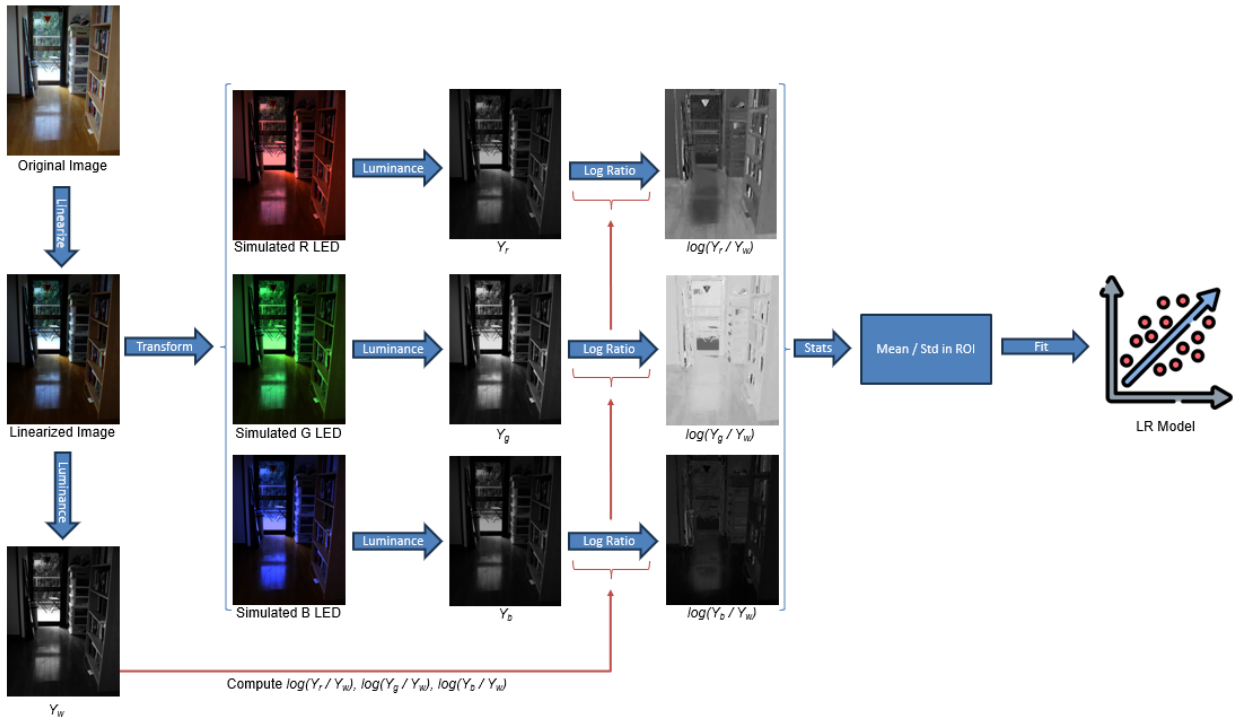

Fig. 3. Overview of the proposed illumination-ratio pipeline. A linearized input image is diagonally transformed to simulate red, green, and blue illuminants. Luminance maps are computed and converted into log-ratio images relative to the white luminance. Mean or standard-deviation statistics within the ROI are extracted from these ratios and used as features for linear regression to predict CoF.

4.2 Baseline Shading Features

Table 1 reports the RMSE of the three shading-based baselines proposed by Brandão et al. [1]. ShadMax and ShadStd are computed from the Gaussian-smoothed log-luminance shading estimate, and GradMu is the mean gradient magnitude in the ROI computed from the luminance image, as described in Section 3.

TABLE 1
Baseline shading features on the GTF dataset (10-fold RMSE)

Method	RMSE
ShadMax	0.173
ShadStd	0.177
GradMu	**0.172**

The results show that GradMu slightly outperforms the shading statistics, resulting in an RMSE of 0.172 compared to 0.173 and 0.177 for ShadMax and ShadStd. These values are consistent with the trends reported in [1] and form our baseline for subsequent comparisons.

4.3 Illumination Ratio Features

We next evaluate the proposed cross-illumination ratio features. For each image we generate luminance maps under simulated red, green, and blue illuminants, compute log-ratio images relative to the original white light luminance, and extract either the mean or standard deviation of these log-ratios within the ROI. We consider all combinations of resulting images as listed in Table 2.

TABLE 2
Illumination-ratio features (10-fold RMSE). Mean and Std denote summary statistics of log-ratios within the ROI.

Ratios Used	Mean RMSE	Std RMSE
$log(Y_r/Y_w), log(Y_g/Y_w), log(Y_b/Y_w)$	0.168	0.195
$log(Y_r/Y_w), log(Y_g/Y_w)$	0.165	0.188
$log(Y_r/Y_w), log(Y_b/Y_w)$	0.167	0.179
$log(Y_g/Y_w), log(Y_b/Y_w)$	0.170	0.181
$log(Y_r/Y_w)$	**0.163**	0.177
$log(Y_g/Y_w)$	0.166	0.175
$log(Y_b/Y_w)$	0.169	0.172

Table 2 summarizes the RMSE for different choices of computed images and statistic. Several trends are apparent. First, using mean log-ratios already improves upon all shading-based baselines. The best-performing configuration is the mean of the red over white log-ratio ($log(Y_r/Y_w)$), which achieves an RMSE of 0.163, an improvement of 0.009 over GradMu (about 5% relative). Adding additional features does not further reduce error, and in some cases slightly degrades performance, suggesting that the most informative signal is concentrated in the red over white ratio.

In contrast, features based on the standard deviation of the log-ratio images perform worse, with RMSE values from 0.172 to 0.195. This indicates that global variation in the ratios is a noisy cue, whereas the mean ratio provides a more stable descriptor of the surface's apparent reflectance under the simulated illuminants.

Overall, the results demonstrate that a single scalar feature, the mean of red over white log-ratio in ROI, is suf-

ficient to consistently outperform all three baselines, while remaining extremely cheap to compute.

4.4 Comparison with DINOv2 Learned Features

To place our handcrafted ratio features in the context of modern deep representations, we compute DINOv2 ViT-B/14 embeddings for each cropped ROI and train a ridge-regression model on top of these features. We also evaluate a hybrid model that concatenates the DINOv2 embedding with our best illumination-ratio feature ($log(Y_r/Y_w)$ mean).

TABLE 3
Comparison with DINOv2 features (10-fold RMSE).

Method	RMSE
DINOv2 (ViT-B/14)	0.165
DINOv2 (ViT-B/14) + $log(Y_r/Y_w)$ mean	0.165

As shown in Table 3, the DINOv2 baseline achieves an RMSE of 0.165, which is better than the shading baselines, but slightly worse than our best performing lag-ratio feature. Interestingly, augmenting DINOv2 with the Y_r/Y_w mean feature does not change performance (the RMSE remains 0.165). This suggests that the DINOv2 representation already encodes information closely related to cross-illumination ratios, so explicitly adding the ratio does not provide additional benefit.

At the same time, the fact that our single scalar $log(Y_r/Y_w)$ feature slightly outperforms DINOv2 on this dataset is noteworthy: it indicates that for this particular controlled setting, carefully designed illumination-ratio cues can exceed the performance of a large self-supervised vision transformer, while being orders of magnitude cheaper to compute.

4.5 Residual Analysis

To better understand where the proposed ratio-based model succeeds and fails, we analyze residuals for the $log(Y_r/Y_w)$ mean feature. For each fold we collect the predicted CoF values and residuals $r_i = y_i - \hat{y}_i$ on the test images, then aggregate them across all 10 folds. Figure 4 shows a scatter plot of residuals versus predicted CoF values for the combined set.

Most predictions lie within an error band of roughly ± 0.2 around zero, indicating that the linear model captures much of the variation in CoF across the dataset. However, there are several notable outliers, with residuals as large as $+0.6$ or -0.4. These correspond to surfaces for which the visual appearance under the available illumination is a poor proxy for friction; for example, cases where a very glossy surface turns out to be surprisingly high-friction, or where a matte surface has unexpectedly low friction. Lastly,the errors tend to be larger in certain ranges of predicted CoF, suggesting that a more flexible regression model or additional features might further improve performance.

In summary, the experimental results show that the proposed illumination-ratio feature provides a modest but consistent improvement over the shading-based baselines of Brandão et al. [1], and performs slightly better than a strong DINOv2 deep-feature baseline, while remaining extremely simple and computationally efficient.

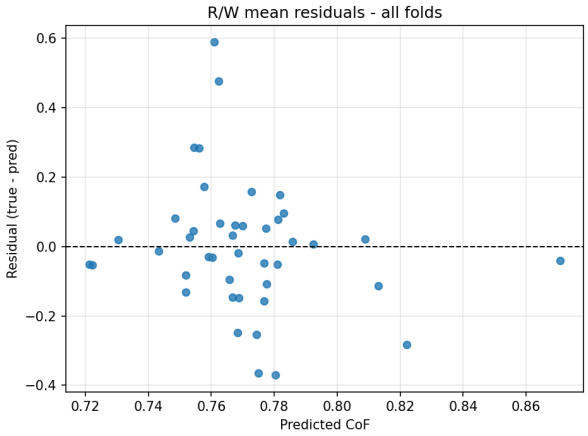

Fig. 4. Residuals versus predicted CoF for the $log(Y_r/Y_w)$ mean model, aggregated over all 10 folds. Most points lie within ± 0.2, with a few larger positive and negative outliers.

5 Conclusion and Future Work

We investigated whether simple illumination-ratio features derived from von Kries transforms can improve prediction CoF from a single RGB image. Using the GTF dataset of Brandão et al. [1], we reimplemented their shading-based baselines (ShadMax, ShadStd, GradMu) and introduced a lightweight pipeline that linearizes sRGB, simulates colored illuminants via per-channel gains, and computes log-luminance ratios between simulated red, green, blue, and white lighting. Simple statistics of these ratio images over the region of interest provide features for linear regression, and we compared them against both the original baselines and a DINOv2 deep-feature regression model.

Our experiment results show that a single scalar feature (the mean of the red over white log-luminance ratio) achieves an RMSE of 0.163 on GTF, outperforming all shading baselines (RMSE 0.172 to 0.177) and slightly outperforms a DINOv2 ViT-B/14 ridge-regression baseline (RMSE 0.165). Residual analysis indicates that most predictions lie within ± 0.2 CoF units of the ground truth, with a small number of outliers corresponding to visually ambiguous materials. Interestingly, augmenting DINOv2 features with our ratio feature does not reduce error, suggesting that modern self-supervised representations already encode information closely related to cross-illumination ratios.

This study has limitations: it is restricted to the controlled GTF dataset [1] with known regions of interest; it models colored illumination using a simple diagonal transform motivated by color constancy theory [3]; and it relies on linear regression, which may underfit more complex relationships between appearance and friction. Future work includes evaluating ratio features on robot-collected data with automatic ROI detection, exploring slightly richer but still lightweight regression models, and designing compact active-illumination setups that approximate the rich reflectance measurements of Zhang et al [2]. Ultimately, integrating friction prediction and its uncertainty into full locomotion and planning pipelines remains an important step toward safer, more agile robotic systems.

ACKNOWLEDGMENTS

The authors would like to thank Professor Lindell for his guidance, feedback, and support throughout this project and for providing the CSC2529 course materials that made this work possible. Lastly, the authors acknowledge that chat-GPT was used to help reformat some parts of this paper.

REFERENCES

[1] M. Brandão, K. Hashimoto, and A. Takanishi, "Friction from vision: A study of algorithmic and human performance with consequences for robot perception and teleoperation," in *Proc. IEEE-RAS Int. Conf. Humanoid Robots (Humanoids)*, Nov. 2016, pp. 428–435.

[2] H. Zhang, K. Dana, and K. Nishino, "Friction from reflectance: Deep reflectance codes for predicting physical surface properties from one-shot in-field reflectance," in *Lecture Notes in Computer Science*, 2016, pp. 808–824.

[3] G. D. Finlayson, M. S. Drew, and B. V. Funt, "Diagonal transforms suffice for color constancy," in *Proc. IEEE Int. Conf. Computer Vision (ICCV)*, Nov. 1993, pp. 164–171.

[4] M. Oquab, T. Darcet, T. Moutakanni, H. Vo, M. Szafraniec *et al.*, "Dinov2: Learning robust visual features without supervision," *arXiv preprint arXiv:2304.07193*, 2023.

[5] X. Cao *et al.*, "Stereoscopic flash and no-flash photography for shape and albedo recovery," in *Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition (CVPR)*, Jun. 2020, pp. 3427–3436.