Visual Robustness Benchmark for Visual Question Answering (VQA)

Supplementary Material

A. Visual Corruption Function Details

A.1. Arithmetic Noise

Arithmetic noise modifies the image by performing arithmetic operations *e.g.* addition, multiplication, and negation on all of the color channels. A subcategory of arithmetic noise is **Additive Noise** which adds a particular value coming from a distribution $\mathcal{D}$ to every pixel in the image. Additive noise is implemented in the form of **Gaussian Noise** and **Poisson Noise**. Gaussian noise appears under low light conditions [28] and is one of the most common types of noise in telecommunications and digital image [6]. Poisson noise or shot noise occurs due to the nature of light behaving as a quantized particle [26].

To define additive noise, we first define the random variable X as $X_n \sim N(\mu, \sigma^2)$ where the probability distribution N is defined as $f(x) = \frac{1}{\sigma\sqrt{2\pi}}e^{-\frac{(x-\mu)^2}{2\sigma^2}}$ and $X_p \sim P(\lambda)$ where the probability distribution P is defined as $f(x) = \frac{\lambda^x}{x!}e^{-\lambda}$. The transformation function for additive noise can be generalized as $\mathcal{T}(r) = r + Y$ where $Y = X_n$ or $Y = X_p$ for Gaussian and Poisson noise respectively. The severity levels are defined by changing the parameter values of the aforementioned probability distributions.

Another subcategory of arithmetic noise is **Multiplicative Noise** implemented in the form of **Speckle Noise** which can also be generalized as $\mathcal{T}(r) = r + r \cdot X_n$ where X_n is the random variable $X_n \sim N(\mu, \sigma^2)$. Speckle noise is a common occurrence in medical and radar images [49]. **Color Inversion**, a common digital image processing operation, performs subtraction *i.e.* $\mathcal{T}(r) = r_{max} - r$. The deterministic function has a single severity level and can also be considered as an image attribute transformation function (Appendix A.3). The reader should note that the value of $\mathcal{T}(r)$ might fall outside the range $[r_{min}, r_{max}]$ and requires clamping.

A.2. Value Assignment Noise

As the name suggests, the value assignment noise has a probability p of assigning a particular value to a pixel, *i.e.* $\mathcal{T}(r) = k$, on all the color channels. This noise is primarily implemented in the form of **Impulse Noise** which is typically one of the two types - bipolar impulse noise, commonly known as **Salt and Pepper Noise**, and **Random Valued Impulse Noise**. Salt and pepper noise takes one of two values, typically between the maximum intensity value r_{max} and the minimum intensity value r_{min} – each with an equal probability p of occurrence. Random-valued impulse noise takes a particular value from a range of values, typically $[r_{min}, r_{max}]$, and follows a uniform distribution for the probabilistic occurrence of the values. A defective camera sensor might cause impulse noise during capturing and transmitting the image [6, 48]. Another form of value assignment can take place in the form of **Thresholding** *i.e.* the pixel will be assigned a binary value based on exceeding or subceeding a particular threshold value, r_{thresh}. **Binary Thresholding** is defined as $\mathcal{T}(r) = r_{max}$ if $r > r_{thresh}$, otherwise, $\mathcal{T}(r) = r_{min}$.

A.3. Image Attribute Transformation

Image attributes *e.g.*, brightness, saturation, contrast, color properties, etc, are often modified to enhance the visual quality of the image [21]. To modify the **Brightness**, we transform the image from the RGB color model to the HSV color model and add a positive or negative constant to the *value* channel of the HSV image to increase or decrease the brightness. The function can be defined as $\mathcal{T}(v) = v + c$ where v represents the value of the **value** channel and c represents the additive constant. In real-life scenarios, lighting effects, luminance adjustment in digital displays, photographic effects, and other factors can cause an image to appear brighter or darker. By simulating these effects using the brightness function, our framework can test the visual robustness of VQA models under varying lighting and display conditions.

Saturation refers to the purity of the colors in an image and can be used to enhance the quality of the image *i.e.* the image will look visually appealing to a human observer [21]. However, oversaturation might make the image look artificial to an observer and undersaturation might produce washed-out effects that can adversely affect the image quality. Changing the saturation is common in digital image processing to make the image look aesthetically pleasing or to reveal seemingly unseen features [13]. Saturation is changed by transforming the image from RGB to HSV color model, followed by modifying the *saturation* channel value by multiplying and adding constants *i.e.* $\mathcal{T}(v) = v \cdot c_1 + c_2$ where c_1 and c_2 represents the multiplicative and additive constants respectively which are set based on the severity of the noise.

Contrast refers to the difference in color intensity values between different parts of the image *i.e.* how well the details of an image are distinguishable [21]. An image having a good level of contrast is more appealing to a viewer as it sets clear boundaries between various color intensities. On the contrary, low contrast creates difficulty in differentiating the details and hence, producing washed-out effects. Contrast enhancement is a common image-processing tech-

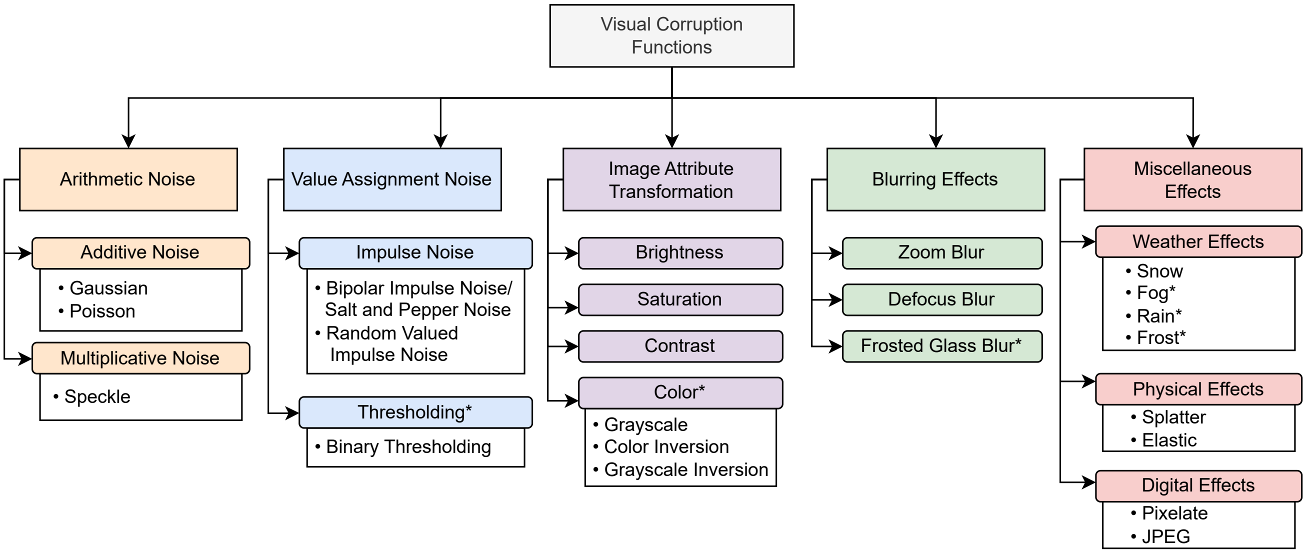

Figure A.1. Comprehensive taxonomy of the visual corruption functions introduced in our work. The functions can be broadly categorized into five main classes similar to [36] which are further divided into multiple sub-classes, providing a detailed overview of the various types of realistic corruptions that can affect the quality of the image. * indicates that the corruption effects are included in our framework results of the corruption effects were not included in our work.

nique applied to spatial, frequency, and wavelet domains using contrast stretching, histogram equalization, etc. The contrast transformation is defined as, $\mathcal{T}(r) = (r-\mu)\cdot c+\mu$ where μ represents the average pixel intensity and c represents the multiplicative constant. Similarly to arithmetic noise, the outputs of all image transformation functions are clamped.

In real-world applications, grayscale images are prevalent due to constraints on representing the color information of a digital image. Several systems such as medical imaging, document scanning, and security work with grayscale images. On the other hand, systems like night vision, medical imaging, astronomy, etc. use color-inverted images. **Grayscale** can be categorized as a transformation function that modifies the color property of the image. Grayscale simply averages the intensity values over the color channels *i.e.* $\mathcal{T}(r) = \mu^C$ where μ^C represents the average pixel intensity over the *color* channel. **Color Inversion**, previously described as arithmetic noise, can be classified as an image attribute transformation function since it modifies the color property of an image. **Grayscale Inversion** is simply the combination of grayscale and color inversion; defined as $\mathcal{T}(r) = r_{max} - \mu_C$.

A.4. Blurring Effects

Blurring effects are produced by convolving with an averaging filter and can be mathematically described as $\mathcal{T}(\mathcal{I}) = \mathcal{I} * K$ where $\mathcal{I}$ represents the digital image and K represents the kernel and convolution operation for a 2D image is defined as

$$(\mathcal{I} * K)(x,y) = \sum_{i=0}^{M-1}\sum_{j=0}^{N-1} \mathcal{I}(i,j)\cdot K(x-i,y-j)$$

While Gaussian blur and median blur are the most common blurring functions, we shall define a few other blurring functions that have common real-life applications. **Defocus Blur** performs channel-wise convolution, and the function is defined as $\mathcal{T}(\mathcal{I}) = \mathcal{I} * K_r$ where K_r is a disk kernel with radius r that varies across severity levels. Defocus blur replicates the blurring effect in cameras when the subject is out of focus. **Zoom Blur** occurs due to rapid camera motion towards an object and **Frosted Glass Blur** imitates the appearance of an object while looking through frosted glass. Most of these effects do not have strict definitions and follows the implementation by [28, 36].

A.5. Miscellaneous Effects

Apart from the previous transformation functions, weather effects can impose a particular weather condition on an image. At the time of writing this paper, our framework includes the **Snow Effect** only but we wish to include other effects like fog, frost, rain, and clouds in the future. We produce the snow effect by creating a snow layer following the normal distribution, then applying the zoom operation, followed by thresholding and motion blur. We use a blending function on the input image and a scaled grayscale version of this image, then add the snow layer and the rotated snow layer to the image to generate the final output of the snow effect.

Some transformation functions try to create **Physical Effects** on the images. The **Splatter Effect** makes the image look like it has been splattered by paint or any form of liquid. The **Elastic Effect** simulates the effect of stretching or wrapping the image. Finally, we included a couple of transformation functions that replicate digitization effects. Digital images are discrete approximations of analog signals, thus various artifacts may remain from the conversion process. The **Pixelate Effect** is a visual effect that creates a mosaic-like appearance, similar to visible image pixels appearing due to lower resolutions, by downsampling and upsampling the image using bilinear interpolation. Pixelation is commonly used for stylistic purposes and censorship. **JPEG Compression Effect** tries to emulate the loss of image information due to JPEG compression [58].

B. Additional Evaluation Metric Details

Nomenclature

General

$v, \mathcal{V}, V$	Model, Set of Models, Number of Models
$c, \mathcal{C}, C$	Corruption, Set of Corruptions, Number of Corruptions
$l, \mathcal{L}, L$	Severity Level, Set of Severity Levels $\{1,2,3,4,5\}$, Number of Severity Levels

Accuracy Metrics

$A_{v,c,l}$	Accuracy for model v, corruption c, and severity level l
$A_{v,0}$	Base Accuracy for model v. Can be rewritten as $A_{v,c,0}$
$A_{v,c}$	Average Accuracy for model v and corruption c
A_v	Corruption-average Accuracy for model v
A_c	Model-average Accuracy for corruption c
A_v^{rel}	Relative Accuracy Drop for model v
A_c^{rel}	Relative Accuracy Drop for corruption c

Robustness Metrics

$E_{v,c,l}$	Error for model v, corruption c, and severity level l
$E_{v,0}$	Base Error for model v,. Can be rewritten as $E_{v,c,0}$
$\mathcal{M}_{v,c}$	Generalized Severity Aggregation Error Metric for model v and corruption c
$\mathcal{M'}_{v,c}$	Non-scaled Severity Aggregation Error Metric for model v and corruption c
$\mathbb{M}$	Set of Severity Aggregation Error Metrics $\{\mathcal{F}, \mathcal{R}, \rho, \mu, \delta\}$
$\mathcal{M}_v$	Corruption Aggregation Error Metric for model v
$\mathcal{M}_c$	Model Aggregation Error Metric for corruption c
$\mathcal{F}$	First-Drop
$\mathcal{R}$	Range of Error
ρ	Error Rate
μ	Average Error
δ	Average Difference of Corruption Error

Visual Robustness Error

$W_{\mathcal{M}}$	Weight assigned to Metric $\mathcal{M}$
$p_{\mathcal{M}}$	Preference Score assigned to Metric $\mathcal{M}$
VRE_v	Visual Robustness Error for model v
VRE_c	Visual Robustness Error for corruption c

B.1. First-Drop Details

The rationale behind using *relative difference* instead of *difference* is that the difference between level-1 and level-0 errors will depend on the model's base accuracy. The relative difference does not have such a dependency and can better capture the error when the model is introduced to corruption effects while encapsulating the variations and decoupling the base accuracy from the equation.

The word *drop* signifies the drop or decrease in accuracy due to introducing the visual corruption. The term might be misleading since, when calculating robustness error, the addition of corruption increases the error. Higher first-drop scores for a model-corruption pair indicate that the model is more prone to lower levels of that particular corruption.

B.2. Range of Error Details

A clear distinction is made between the maximum error value and the error value at the highest severity level in Eq. (11) as the error value at the highest severity level does not imply that it has the maximum error value.

B.3. Robustness Evaluation Scenarios using VRE

Scenario 1. We assume the VQA model is deployed in an environment where minor levels of visual corruption occur *e.g.* slight changes in weather or lighting conditions. This scenario is well-suited for the first-drop evaluation metric, which captures robustness at lower severity levels. The average error, which provides an overall estimation of performance, can also be used. VRE can be tuned by assigning more weight to the first drop and average error metrics.

Scenerio 2. We consider that the VQA model is experiencing corruption effects that gradually increase in intensity *e.g.* the images are getting blurrier or brighter over time. In this case, the range of error and error rate can jointly evaluate the robustness. VRE can be similarly tuned to prioritize the aforementioned metrics.

Model	Lvl	Noise				Blur		Weather	Image Attrribute			Physical		Digital	
		Shot	Gaus	Imp	Spec	Defo	Zoom	Snow	Brig	Cont	Sat	Elas	Spl	Pix	JPEG
ViLT	0	**0.287**													
	1	0.326	0.290	0.297	0.290	0.295	**0.392**	0.341	0.287	0.290	0.290	0.292	0.286	**0.280**	0.282
	2	0.374	0.304	0.314	0.293	0.307	**0.439**	0.395	0.291	0.298	0.307	0.296	0.317	**0.281**	0.285
	3	0.460	0.329	0.334	0.321	0.332	**0.472**	0.424	0.301	0.317	0.323	0.305	0.348	**0.284**	0.290
	4	**0.526**	0.369	0.380	0.336	0.359	0.503	0.458	0.312	0.407	0.356	0.329	0.370	**0.291**	0.303
	5	**0.569**	0.436	0.437	0.364	0.387	0.525	0.475	0.328	0.567	0.359	0.366	0.408	0.340	**0.322**
BLIP	0	**0.218**													
	1	0.281	0.238	0.253	0.236	0.254	**0.337**	0.286	**0.228**	0.237	0.240	0.233	0.233	0.238	0.239
	2	0.328	0.254	0.270	0.244	0.276	**0.377**	0.322	**0.236**	0.248	0.261	0.251	0.262	0.248	0.251
	3	0.410	0.283	0.287	0.275	0.308	**0.411**	0.325	**0.246**	0.267	0.278	0.270	0.292	0.257	0.258
	4	**0.471**	0.324	0.332	0.299	0.338	0.441	0.358	**0.260**	0.320	0.305	0.280	0.299	0.276	0.285
	5	**0.533**	0.389	0.394	0.329	0.367	0.461	0.373	**0.275**	0.414	0.315	0.366	0.332	0.349	0.316
VLE	0	**0.229**													
	1	0.311	0.250	0.261	0.245	0.271	**0.366**	0.266	**0.230**	0.235	0.238	0.242	0.233	0.236	0.248
	2	0.387	0.271	0.290	0.260	0.296	**0.410**	0.312	**0.233**	0.241	0.242	0.249	0.255	0.238	0.267
	3	**0.499**	0.324	0.323	0.309	0.347	0.425	0.314	**0.241**	0.253	0.265	0.287	0.275	0.256	0.275
	4	**0.567**	0.400	0.404	0.356	0.392	0.468	0.338	**0.249**	0.307	0.286	0.291	0.278	0.280	0.316
	5	**0.606**	0.487	0.478	0.396	0.420	0.489	0.350	**0.267**	0.431	0.301	0.411	0.313	0.387	0.373
PNP	0	**0.352**													
	1	0.585	0.575	0.580	0.575	0.579	**0.603**	0.590	**0.570**	0.576	0.576	0.572	0.574	0.573	0.572
	2	0.599	0.578	0.583	0.576	0.584	**0.612**	0.603	0.574	0.580	0.590	0.578	0.585	**0.574**	0.575
	3	**0.626**	0.585	0.586	0.586	0.591	0.621	0.606	0.578	0.586	0.593	0.582	0.590	**0.575**	0.577
	4	**0.650**	0.599	0.601	0.588	0.599	0.629	0.615	0.580	0.606	0.607	0.583	0.598	**0.580**	0.582
	5	**0.681**	0.616	0.615	0.601	0.611	0.642	0.619	**0.588**	0.638	0.610	0.608	0.608	0.599	0.590
LLaVA-7B	0	**0.217**													
	1	0.298	0.246	0.262	0.256	0.275	**0.301**	0.296	0.217	0.226	0.231	0.225	0.229	**0.217**	0.218
	2	0.341	0.267	0.280	0.264	0.291	**0.348**	0.319	0.221	0.239	0.245	0.234	0.257	**0.217**	0.220
	3	**0.396**	0.281	0.303	0.273	0.313	0.372	0.322	0.230	0.249	0.271	0.249	0.291	**0.222**	0.225
	4	**0.453**	0.319	0.341	0.312	0.338	0.413	0.347	0.243	0.315	0.294	0.277	0.299	**0.229**	0.231
	5	**0.542**	0.374	0.407	0.346	0.369	0.452	0.361	**0.259**	0.397	0.301	0.349	0.327	0.283	0.267
LLaVA-13B	0	**0.202**													
	1	0.291	0.241	0.258	0.245	0.271	**0.300**	0.283	**0.203**	0.212	0.223	0.220	0.220	0.208	0.204
	2	0.333	0.265	0.279	0.259	0.293	**0.348**	0.304	**0.205**	0.225	0.238	0.223	0.255	0.209	0.209
	3	**0.378**	0.282	0.305	0.267	0.315	0.371	0.310	**0.212**	0.234	0.264	0.239	0.287	0.219	0.215
	4	**0.447**	0.323	0.347	0.304	0.339	0.411	0.341	0.234	0.312	0.291	0.271	0.298	**0.225**	0.229
	5	**0.541**	0.376	0.408	0.336	0.372	0.449	0.358	**0.249**	0.386	0.297	0.338	0.325	0.272	0.258

Table A.1. Error values of the model across various severity levels for different visual corruption effects. **Green** and **red** indicate the minimum and maximum error values of that particular corruption and severity level respectively. The minimum values are observed at only three corruption effects – brightness, pixelate, and JPEG, while the maximum values are observed at zoom blur and shot noise only.

C. Robustness across Question Types

The question category composition of our subsample of the VQAv2 dataset [22] has been explored in fig-A.3, where the "What" type questions are most frequently observed. Following this, most of the questions encountered by the VQA models are asked to classify certain objects or their properties, for instance, "What is", "What color", and "What animal". The next two most encountered question types, "Is" and "Are", ask the model to verify the existence or absence of something in the image. These questions are inherently biased to allude to an object or an action being present in the image itself. Thus, we conclude that the questions from the VQAv2 dataset are not diverse and are somewhat lacking in judging the model's critical thinking capabilities. Questions starting with "How to", "Why is", "Who", and more are rarer while the dataset lacks questions challenging the counting ability of the model *e.g.* questions with "How many".

D. Discussion

D.1. The Necessity of Robustness

A model selected based on high average accuracy or low average error may provide precise and correct predictions under ideal conditions but becomes susceptible to producing erroneous outputs when faced with variations, uncer-

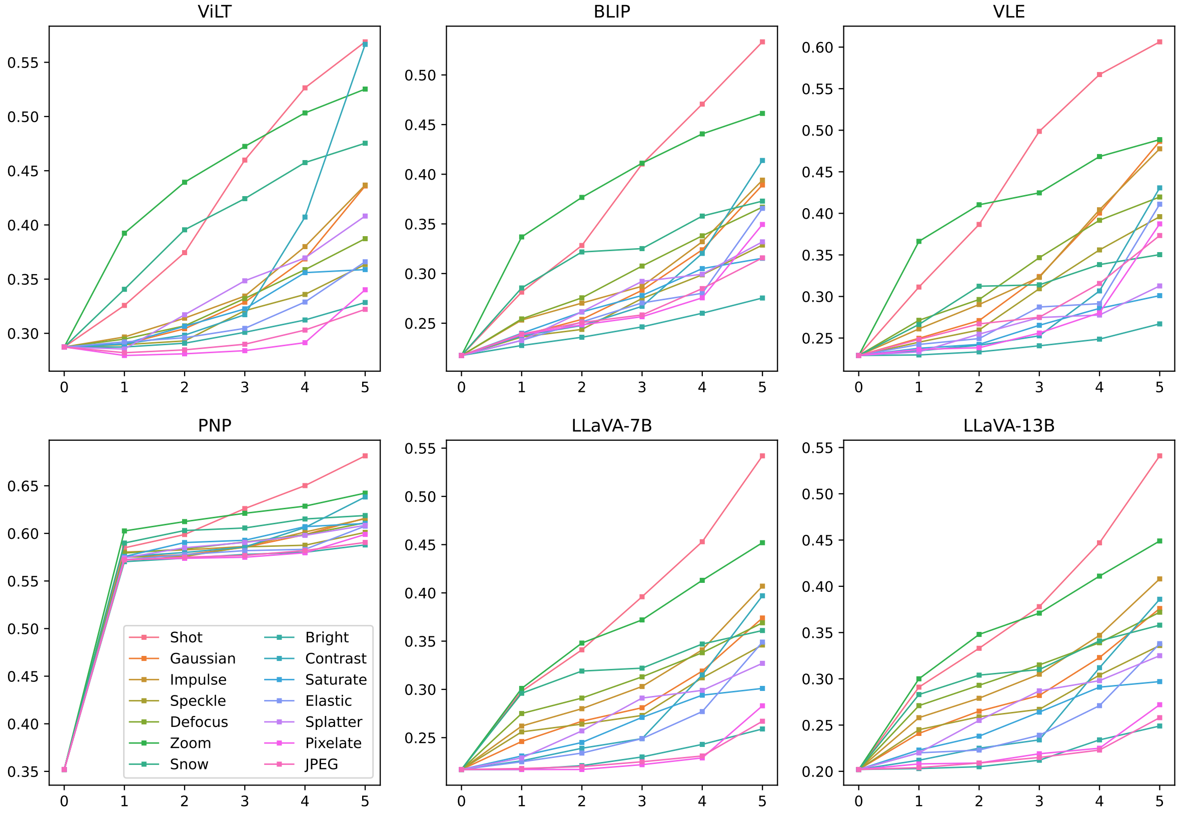

Figure A.2. Error trends over severity levels. The standard VQA models – ViLT [39], BLIP [41], VLE [31], and LLaVA [45] exhibit a somewhat linear rise while the zero-shot VQA model PNP [64] shows an initial sharp rise followed by a linear trend. Logarithmic trend lines are observed by a few corruption functions *e.g*. zoom blur.

Corruption	**Is this**	**What**	**What is**	**Is/are the**	**How many**	**What color is**	**What kind of**
Shot	0.731	0.363	0.312	0.711	0.398	0.557	0.412
Gaussian	0.812	0.454	0.432	0.764	0.466	0.698	0.528
Impulse	0.817	0.445	0.428	0.762	0.453	0.691	0.525
Speckle	0.828	0.476	0.457	0.771	0.483	0.722	0.546
Defocus	0.812	0.444	0.412	0.766	0.442	0.731	0.524
Zoom	0.731	0.369	0.297	0.683	0.350	0.663	0.413
Snow	0.771	0.428	0.389	0.741	0.431	0.656	0.491
Brightness	0.843	0.494	0.489	0.798	0.561	0.718	0.579
Contrast	0.814	0.453	0.433	0.769	0.457	0.659	0.518
Saturation	0.833	0.473	0.465	0.799	0.499	0.597	0.545
Elastic	0.811	0.462	0.451	0.774	0.466	0.758	0.546
Pixelate	0.849	0.480	0.467	0.783	0.486	0.756	0.557
JPEG	0.848	0.481	0.473	0.780	0.480	0.737	0.553
Spatter	0.815	0.460	0.422	0.772	0.479	0.720	0.531

Table A.2. Average accuracy across different types of questions for the visual corruption functions. The models primarily struggle with the "How many" and "What is" questions.

tainties, or adversarial inputs. However, a robust model selected based on multiple aspects exhibits a higher level of resilience and generalization, capable of performing consistently across a wide range of inputs, even in the face of

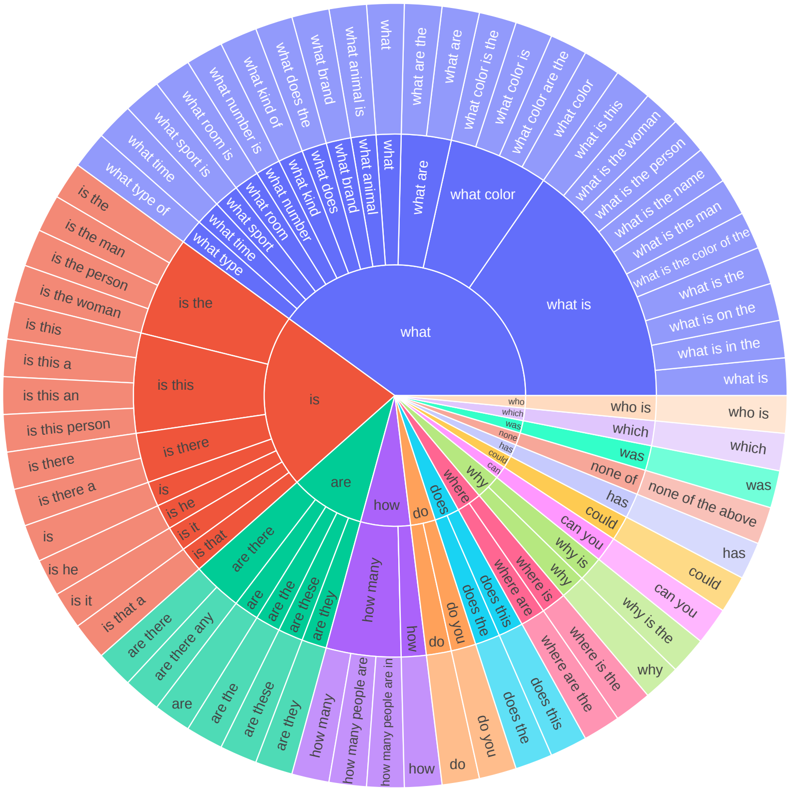

Figure A.3. Breakdown of different question types from the VQAv2 [22] dataset. The most frequent questions start with “What” followed by “Is”, and “Are”.

perturbations or challenging scenarios. The increased robustness might come at the cost of sacrificing accuracy, as the model adopts a more conservative or cautious approach to minimize error.

The trade-off between accuracy and robustness is crucial to consider when developing machine learning models for various applications. Different contexts and use cases may require varying degrees of emphasis on accuracy and robustness. For instance, in safety-critical systems, such as autonomous vehicles or medical diagnosis, robustness takes precedence over accuracy to ensure reliable performance even in uncertain or unpredictable situations. However, in tasks where precision and correctness are paramount, sacrificing some robustness may be acceptable to achieve higher accuracy.

Understanding this trade-off enables researchers and practitioners to make informed decisions when designing models, striking a balance that aligns with the specific requirements and priorities of the given application. It also highlights the need for comprehensive evaluation metrics that consider both accuracy and robustness, providing a more holistic assessment of model performance. As highlighted in Fig. 7, our findings emphasize the delicate interplay between accuracy and robustness in VQA models. Recognizing and managing this trade-off is essential for developing models that align with the desired performance objectives in various real-world scenarios.

D.2. Mislabeling Problem in Grayscale Images and Color Bias

Grayscale images are void of color and hence, the answer to every color-related question on grayscale images should either be unanswerable or a shade of gray. The answers predicted by the model are given full scores as they would match the ground truth color. But grayscaling an image changes the ground truth and hence will require relabeling to prevent inaccurately assessing a model’s performance and robustness. As we did not relabel the grayscale images,

Problem	Questions	Ground Truth	Predictions
Miscolor	What is the bike's color?	blue	black
	What color is the sky?	blue	gray
	What is the color of the soap?	yellow	white
Undercount	How many spoons are there?	2	1
	How many people are there?	5	3
	How many kites are up?	4	3
Misclassify	What is she eating?	sandwich	cake
	What is the weather like?	sunny	cloudy
	What game is this?	baseball	soccer
Blindness	What's on the television?	baby	nothing
	Are the women selling?	yes	no
	What is the cat eating?	cake	nothing
Irrationality	Which bowl has more oranges?	front	right
	What is the man about to do?	run	bat
	What is the man doing?	standing	flying kite

Table A.3. Different types of misprediction problems by ViLT [39] when introduced to the visual corruption effects.

performance related to grayscale images has not been covered in our work.

Fig. A.4 highlights a few color-related questions on grayscale images where ViLT [39] predicted a color, indicating that the model associated colors with shapes or structures in the image. As models were able to predict certain colors on images void of color, we can hypothesize that VQA models exhibit some form of color bias. For instance - if the model sees the gray image of an apple, and is asked "What is the color of the apple?", it will most likely predict *red* since most of the images of apples it was trained on had the color red. Hence, it associated the color red with the shape of the apple. Color bias is caused due to the model's inability to retrieve contextual information from the image as seen in [22].

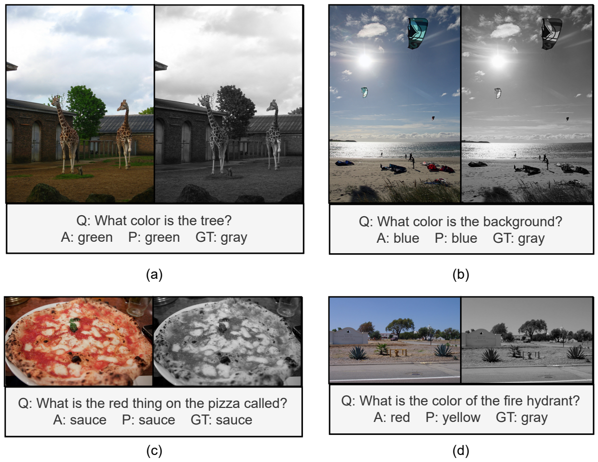
Q: What color is the tree?
A: green P: green GT: gray
(a)
Q: What color is the background?
A: blue P: blue GT: gray
(b)
Q: What is the red thing on the pizza called?
A: sauce P: sauce GT: sauce
(c)
Q: What is the color of the fire hydrant?
A: red P: yellow GT: gray
(d)

Figure A.4. Color-based questions on grayscale images causing mislabeling problems for ViLT [39] and indicating the presence of color bias.

D.3. Zero-shot and Robustness

Experimental results reveal that the Zero-Shot VQA (ZS-VQA) model PNP [64] is more prone to visual corruption effects compared to traditional methods. However, experiments were conducted on a single ZS-VQA model, the subpar robustness performance cannot be generalized for all ZS-VQA models. PNP exhibits a modular architecture where the overall robustness will depend on the individual robustness of each module. The composing modules: image-question matching module, image captioning module, question-answering module, etc, exhibit different levels of visual robustness. Trivially, we can say that the unimodal question-answering module is unaffected by visual noise while the multimodal image-question matching module and image captioning module are both susceptible to visual noise.

The low robustness value of PNP can be loosely associated with the low robustness of its composing modules. If the modules are replaced with more robust counterparts, then PNP might become a more robust model. Low robustness scores for ZS-VQA models might seem counter-intuitive as these models are aimed towards handling unseen or out-of-distribution data [18, 63]. By definition, ZS-VQA models should adapt to different contexts and inputs, making them more resilient to variations and uncertainties. This characteristic is particularly valuable in real-world applications where encountering new or unexpected scenarios is common.

D.4. Unanswerability

Can the corruption functions render some of the questions unanswerable at certain severity levels? Although this is rare, higher corruption intensities can indeed make some questions unanswerable, even by human evaluators. As the VQA models are classifiers, they should predict the correct answer class as "unanswerable" or "nothing". However, similar to grayscale images, this would create a mislabeling problem as the original labels associated with the uncorrupted images were answerable. One plausible solution is manually relabelling the corrupted dataset which can be accurate but costly and impractical.

E. Additional Future Directions

In pursuit of developing a universal robustness evaluation framework, we aim to extend our work by including textual noise, specifically on the input questions. Current literature has explored various forms of textual noise *e.g.* question paraphrasing, semantic error, syntax error [30,35]. Additionally, we propose to simulate typing errors on a physical keyboard [40] by associating a probability distribution with each letter being inserted, repeated, removed, replaced, or exchanged with another letter. For instance, the probability of replacement will depend on the proximity of the other letter to the pivot letter based on the layout of the keyboard.

We plan to incorporate consistency metrics [35] which can be described as an evaluation metric to quantify the model's ability to provide consistent predictions with changes to the input. For instance - for binary classification, if the model predicts 0,1,0,1,0 for five severity levels then it would be deemed inconsistent due to fluctuating predictions. We wish to explore the similarities and differences between consistency and robustness.

Preprocessing the visual or textual input as a form of denoising [33, 54] might mitigate the performance drop due to corruption effects. For a particular modality, the user can opt to use white-box preprocessing *i.e.* processing the input, given the corruption type, or black-box preprocessing *i.e.* processing the input without any prior knowledge of the corruption type. A VQA model utilizing a denoising module might produce better robustness scores than standard approaches.

VQA models can also be trained on noisy data *e.g.* noise textual labels [69]. The noisy data can include corrupted images and textual noise on both questions and answers. Additional explainable AI techniques in VQA [9, 44] can be used to comprehend the processing of visual information by models trained on noisy data during inferences. Training models on grayscale images while retaining the original color labels can help us understand how models perceive shades of grey and whether they associate a specific shade with a particular color.