NEURAL INFORMATION PROCESSING SYSTEMS

HANYANG UNIVERSITY

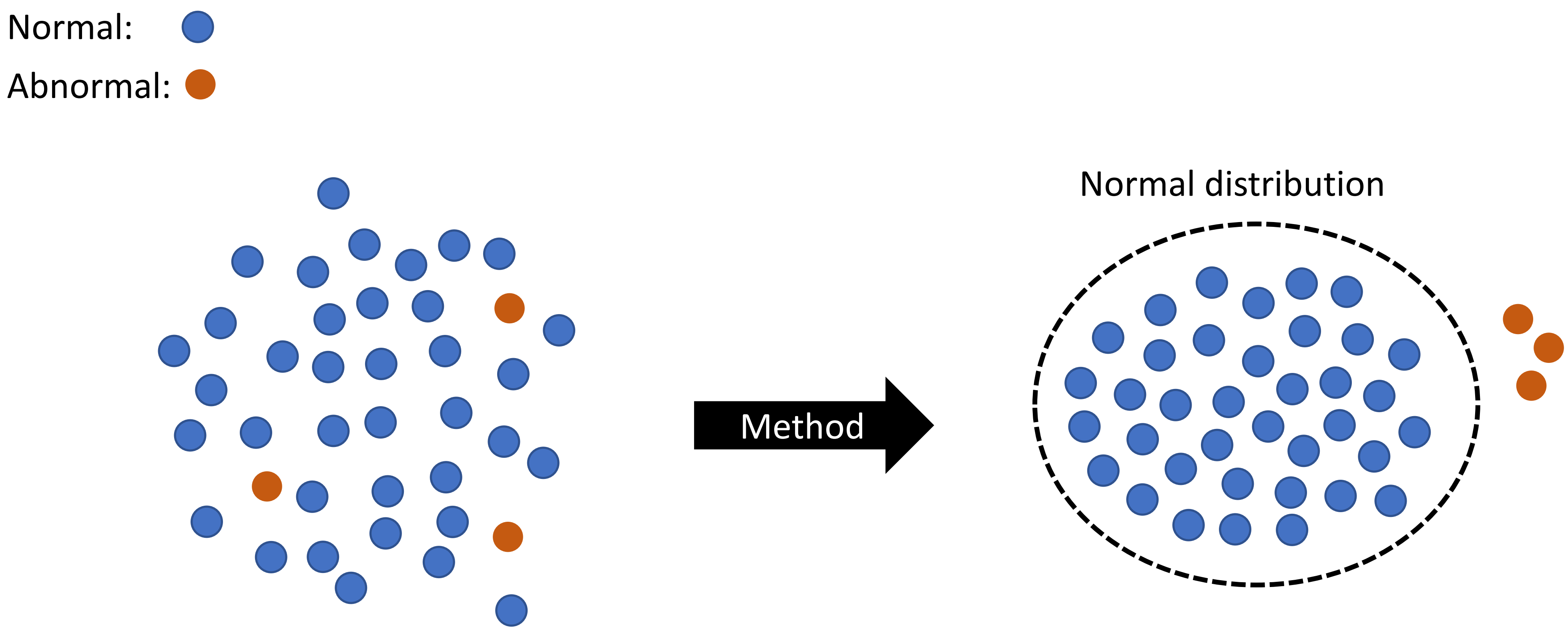
Visual Intelligence Lab.

Semantic-Aware Normalizing Flow for Anomaly Detection and Localization

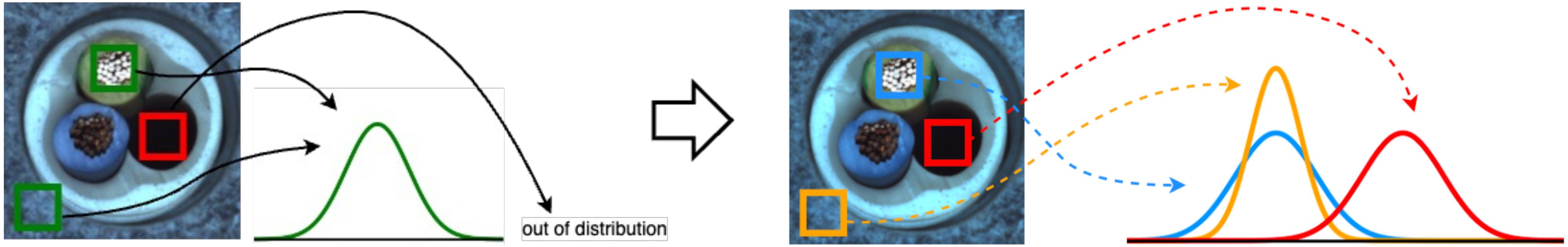
Daehyun Kim[1], Sungyong Baik[2], Tae Hyun Kim[3*]

Visual Intelligence Lab, Hanyang University

Anomaly Detection



Motivation



Anomaly Input　Normal Distribution　Anomaly Input　Estimated Base Distributions

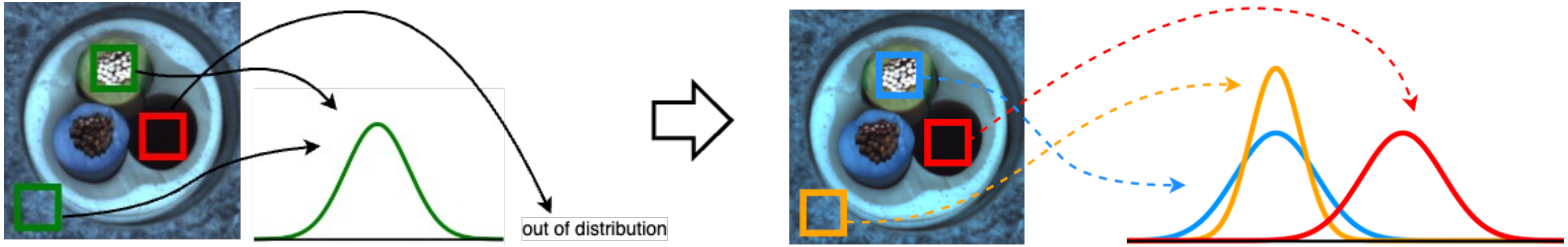
Base distribution of normal data in complex region

Base distribution of normal data in simple region

Base distribution of abnormal region

- Previous works: Transforms normal feature vectors to $N(0,1)$.
- Our work:
  a. Estimate variances of normal feature vector and transform feature vectors to appropriate distributions $N\left(0,\sigma_i^2\right)$.
  b. Like sending normal feature vectors to $N(0,\sigma_i^2)$, added training to send abnormal feature vectors to $N\left(1,\sigma_i^2\right)$ for discriminability.

Motivation



Anomaly Input　Normal Distribution　Anomaly Input　Estimated Base Distributions

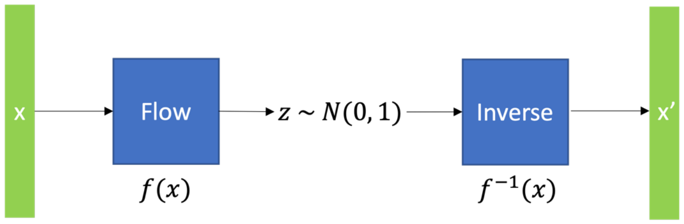
- Previous works: Transforms normal feature vectors to $N(0,1)$.
- Our work:
  a. Estimate variances of normal feature vector and transform feature vectors to appropriate distributions $N\left(0,\sigma_i^2\right)$.
  b. Like sending normal feature vectors to $N(0,\sigma_i^2)$, added training to send abnormal feature vectors to $N\left(1,\sigma_i^2\right)$ for discriminability.

Base distribution of normal data in complex region

Base distribution of normal data in simple region

Base distribution of abnormal region

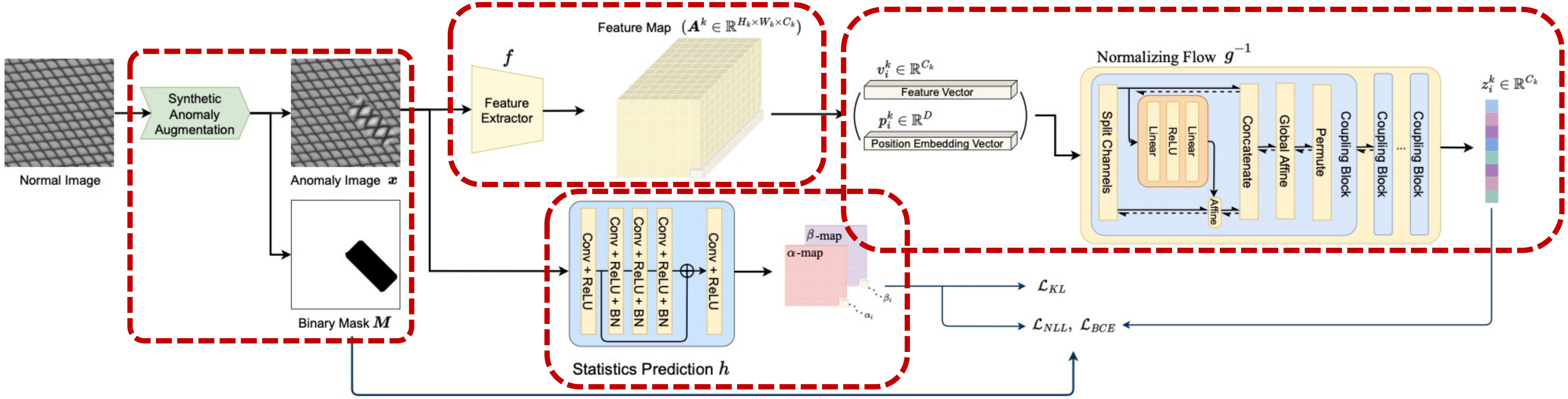
Normalizing Flow



Method

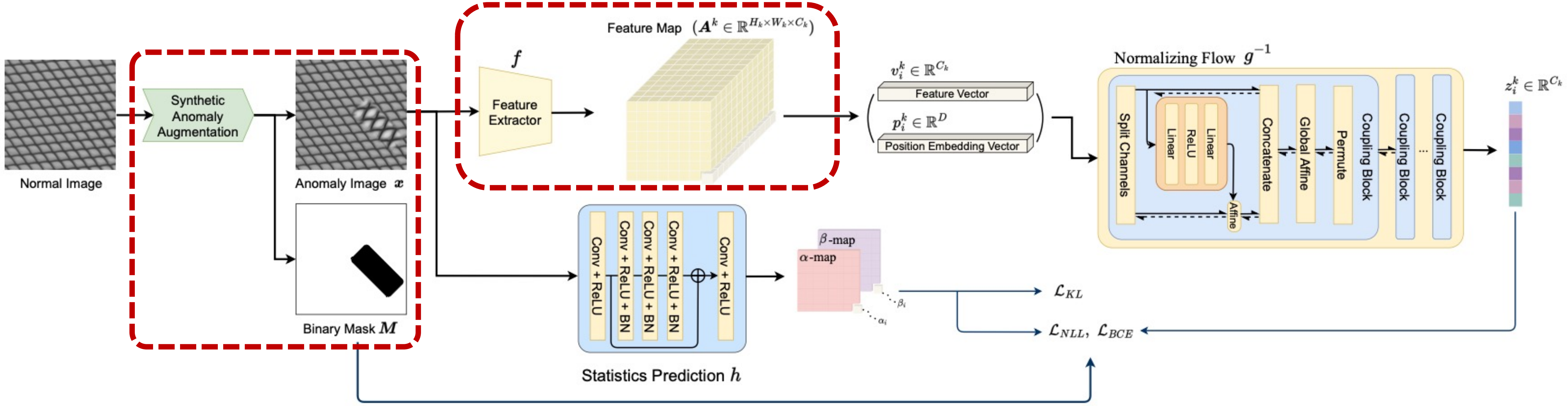
- **Idea 1: Split distributions**
  - Abnormal image augmentation as CutPaste did.
  - By using segmentation masks of pseudo anomalies, can split normal and abnormal feature vectors to parted distributions.
  - $\log p_i = m_i \log p_i^n + (1 - m_i) \log p_i^a$, ($m_i$: 0 for normal and 1 for abnormal)
- **Idea 2: Estimate variances of distributions**
  - Statistic prediction network for inferring variances. *(VDNet Neurips2019)*
  - Assumes the variances of the patches have an inverse-gamma distribution.
  - To find inverse-gamma distribution we predict $\underline{\alpha_i \text{ and } \beta_i}$ and defined <u>a mode value</u> of $IG(\alpha_i, \beta_i)$ as a variance of distribution $N(0, \sigma_i^2)$.

Framework



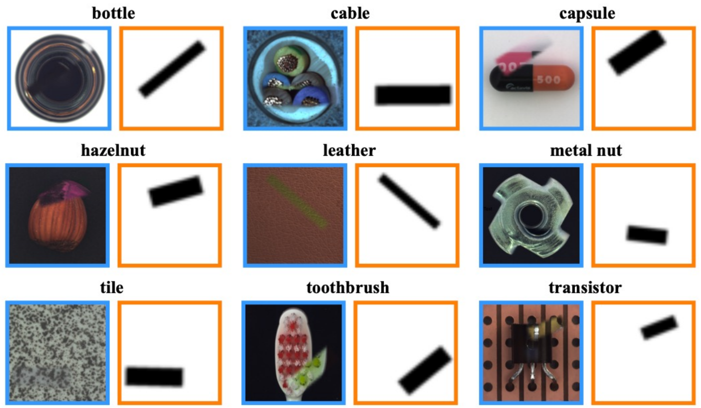
A. **Synthetic Anomaly Augmentation**
B. **Feature Extractor**
C. **Semantic-Aware Normalizing Flow**
D. **Statistic-Aware Base Distribution**

Framework: Synthetic Anomaly Augmentation & Feature Extractor



A. Synthetic Anomaly Augmentation

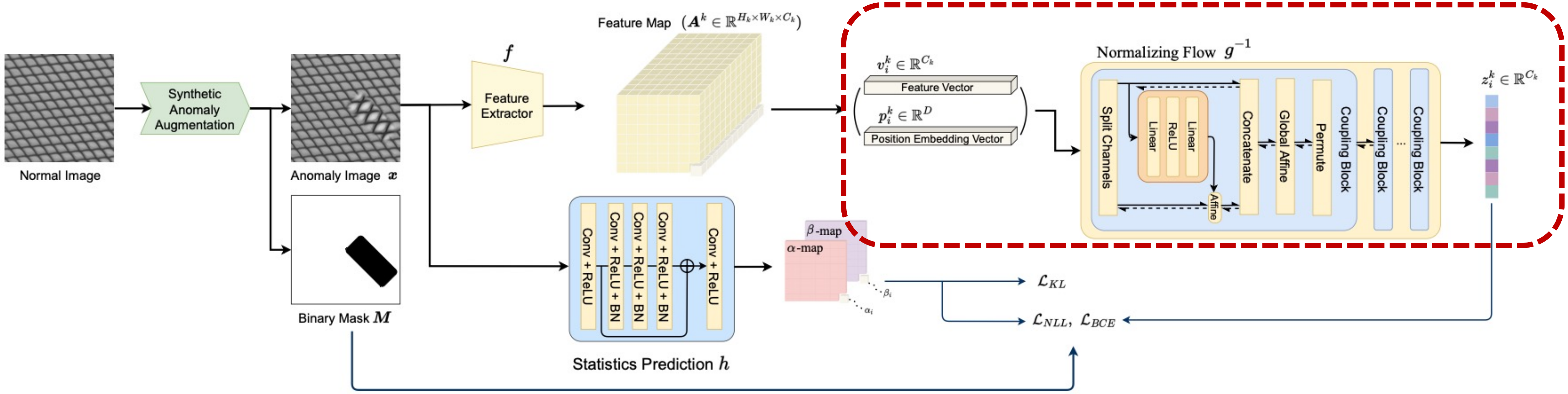
- Synthesize abnormal images to facilitate training of NF. *(CutPaste CVPR2021)*
- Abnormal images usually differ from normal images only in local regions, which are semantically or structurally similar to surrounding normal regions.
- To generate such abnormal data we pasted corrupted patches to normal images.



B. Feature Extractor

- Pre-trained CNN to obtain a 3-level feature pyramid.

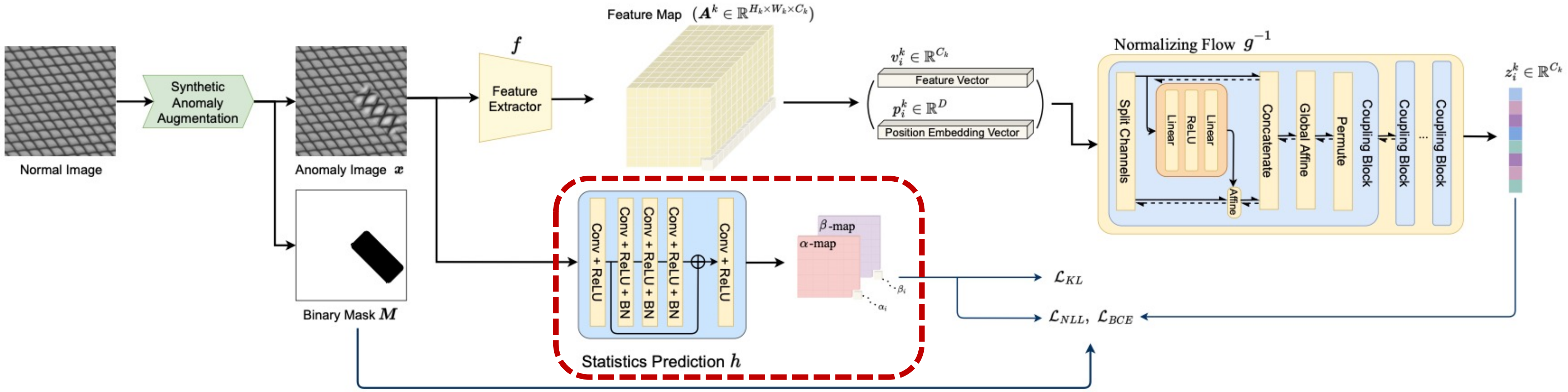
Framework: Semantic-Aware Normalizing Flow



C. Semantic-Aware Normalizing Flow

- Feature vectors can follow complex distributions at each **scale** and **spatial** location of feature map.
- Thus, using a single NF model to map such complex features to a single base distribution can be difficult.
- **Scale:** Employed three independent NF models to handle features across difficult scales as we have 3-level pyramid features and position embedding vector conditioning.
- **Spatial:** Trained NF models to map feature vectors to latent vectors that follow a spatially varying underlying distribution.

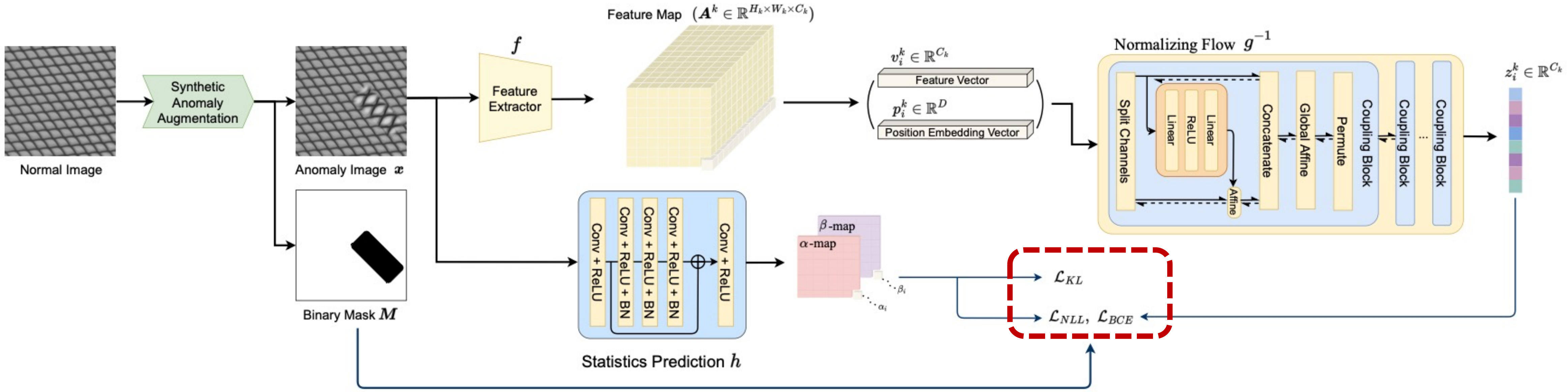
Framework: Statistic-Aware Base Distribution



D. Statistic-Aware Base Distribution

- Employ a lightweight statistics prediction network h to estimate the variances (σ_i^2) for the given feature vector v_i.
- Feature vector will transfer to N(μ_i, σ_i^2).
- Normal region: $\mu_i = 0$, Abnormal region: $\mu_i = 1$
- When samples are non-i.i.d: $\sigma_i^2 \sim IG(\alpha_i, \beta_i)$, IG is a inverse Gamma distribution.

Framework: Loss functions



Loss function & Anomaly score

1. $\boldsymbol{L_{NLL} = \log p_{Z_i}(z_i) - \sum_{l=1}^{L} \log|\det \frac{dg^l}{dz_i^{l-1}}|}$

- $\log p_{Z_i^n}(z_i) = -\frac{1}{2}\log 2\pi - \frac{1}{2}\big(\log\beta_i - \psi(\alpha_i)\big) - \frac{\alpha_i}{2\beta_i}||z_i||_2^2$
- $\log p_{Z_i^a}(z_i) = -\frac{1}{2}\log 2\pi - \frac{1}{2}\big(\log\beta_i - \psi(\alpha_i)\big) - \frac{\alpha_i}{2\beta_i}||z_i - 1||_2^2$
- $\log p_{Z_i}(z_i) = m_i \log p_{Z_i^n}(z_i) + (1 - m_i)\log p_{Z_i^a}(z_i)$

2. $\boldsymbol{L_{KL} = D_{KL}(IG(\alpha_i, \beta_i)\,|\,IG(\alpha, \beta))}$

$$= \sum_{i=1}^{H_k \times W_k}\{(\alpha_i - \alpha)\cdot\psi(\alpha_i) + (\log\Gamma(\alpha) - \log\Gamma(\alpha_i)) +$$

$$\alpha(\log\beta_i - \log\beta)\} + \alpha_i(\frac{\beta}{\beta_i} - 1)$$

3. $\boldsymbol{L_{BCE} = binary\_cross\_entropy(s(z_i), m_i)}$

- $s(z_i) = \frac{p_{z_i^n}(z_i)}{p_{z_i^n}(z_i) + p_{z_i^a}(z_i)}$
- m_i: corresponding binary mask value of augmented abnormal images.

$$L_{total} = L_{NLL} + \lambda_1 \cdot L_{BCE} + \lambda_2 \cdot L_{KL}$$

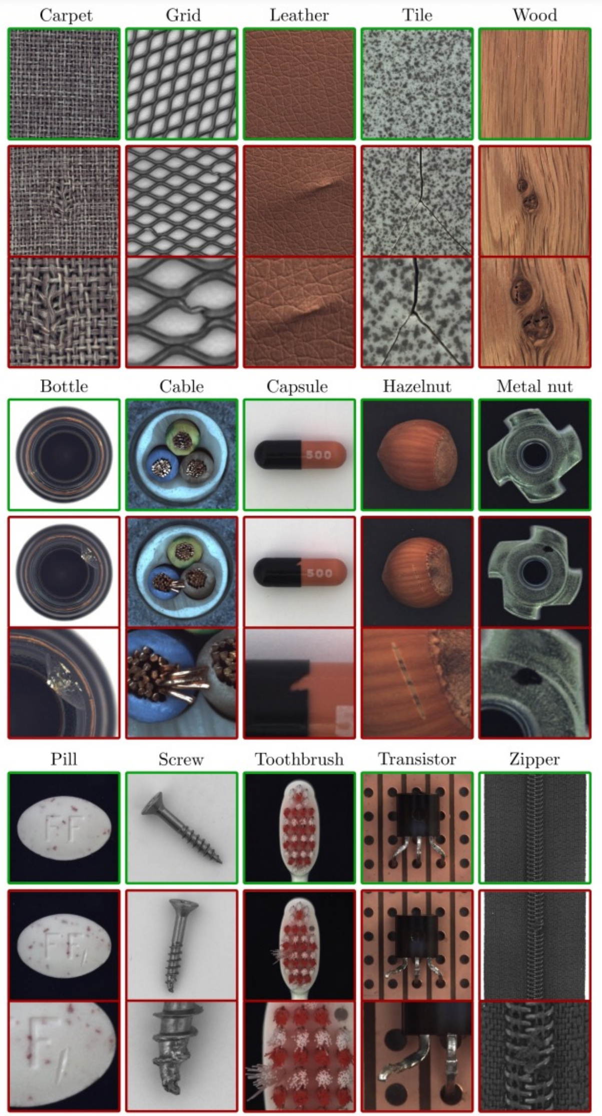
Anomaly Score:

$$= -\frac{1}{2}\log 2\pi - \frac{1}{2}\big(\log\beta_i - \psi(\alpha_i)\big) - \frac{\alpha_i}{2\beta_i}||z_i||_2^2$$

Experiments: Benchmark

MVTec-AD

	Category	# Train	# Test (good)	# Test (defective)	# Defect groups	# Defect regions	Image side length
Textures	Carpet	280	28	89	5	97	1024
	Grid	264	21	57	5	170	1024
	Leather	245	32	92	5	99	1024
	Tile	230	33	84	5	86	840
	Wood	247	19	60	5	168	1024
Objects	Bottle	209	20	63	3	68	900
	Cable	224	58	92	8	151	1024
	Capsule	219	23	109	5	114	1000
	Hazelnut	391	40	70	4	136	1024
	Metal Nut	220	22	93	4	132	700
	Pill	267	26	141	7	245	800
	Screw	320	41	119	5	135	1024
	Toothbrush	60	12	30	1	66	1024
	Transistor	213	60	40	4	44	1024
	Zipper	240	32	119	7	177	1024
	Total	3629	467	1258	73	1888	-



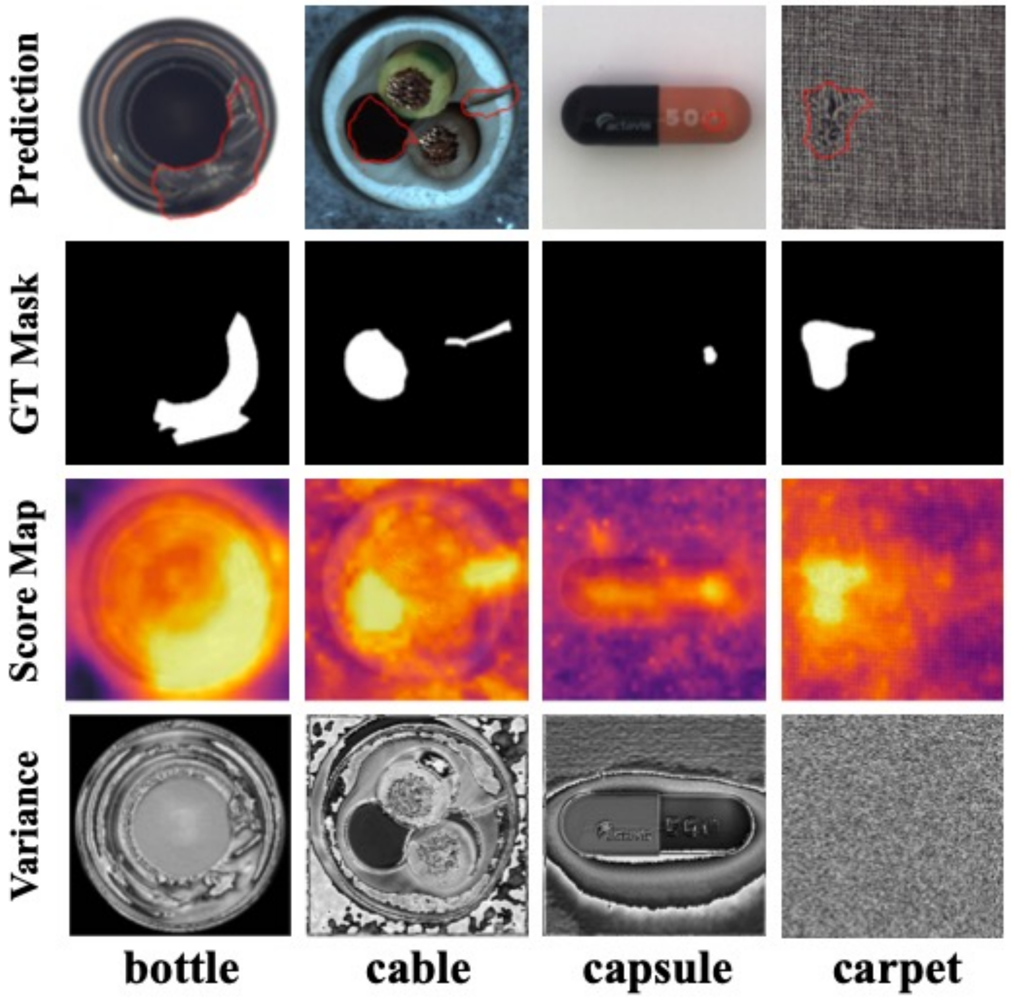
Experiments

Pixel wise

Category	AE_{SSIM} WRN-50	γ-VAE+grad WRN-50	PatchSVDD WRN-50	PaDiM WRN-50	CutPaste WRN-50	CFLOW-AD WRN-50	PatchCore-10 WRN-50	PatchCore-1 WRN-101	**SANFlow** WRN-50	**SANFlow** WRN-101
bottle	93.0	93.1	98.1	98.3	97.6	98.76	98.6	98.6	98.6	**99.1**
cable	82.0	88.0	96.8	96.7	90.0	97.64	98.5	98.4	98.5	**98.8**
capsule	94.0	91.7	95.8	98.5	97.4	98.98	98.9	**99.1**	**99.1**	98.9
carpet	87.0	72.7	92.6	99.1	98.3	99.23	99.1	98.7	99.3	**99.4**
grid	94.0	97.9	96.2	97.3	97.5	96.89	98.7	98.7	98.5	**99.3**
hazelnut	97.0	98.8	97.5	98.2	97.3	98.82	98.7	98.8	**99.2**	99.0
leather	78.0	89.7	97.4	99.2	99.5	99.61	99.3	99.3	99.6	**99.8**
metal nut	89.0	91.4	98.0	97.2	93.1	98.56	98.4	**98.8**	98.5	98.7
pill	91.0	93.5	95.1	95.7	95.7	98.95	97.6	97.8	**99.2**	99.1
screw	96.0	97.2	95.7	98.5	96.7	98.10	**99.4**	99.3	99.0	99.2
tile	59.0	58.1	91.4	94.1	90.5	97.71	95.9	96.1	98.9	**99.1**
toothbrush	92.0	98.3	98.1	98.8	98.1	98.56	98.7	98.8	98.9	**99.2**
transistor	90.0	93.1	97.0	**97.5**	93.0	93.28	96.4	96.4	94.4	95.1
wood	73.0	80.9	90.8	94.9	95.5	94.49	95.1	95.1	96.4	**97.9**
zipper	88.0	87.1	95.1	98.5	99.3	98.41	98.9	98.9	98.9	**99.6**
average	87.0	88.8	95.7	97.5	96.0	97.9	98.1	98.2	98.5	**98.8**

Image wise

Category	GANomaly WRN-50	OCSVM WRN-50	PatchSVDD WRN-50	DifferNet WRN-50	PaDiM WRN-50	CFLOW-AD WRN-50	CutPaste WRN-50	PatchCore-10 WRN-50	PatchCore-1 WRN-101	**SANFlow** WRN-50	**SANFlow** WRN-101
bottle	89.2	99	98.6	99.0	-	**100**	98.2	**100**	**100**	**100**	**100**
cable	75.7	80.3	90.3	95.9	-	97.59	81.2	99.4	99.6	99.4	**99.7**
capsule	73.2	54.4	76.7	86.9	-	97.68	98.2	97.8	98.2	97.7	**98.9**
carpet	69.9	62.7	92.9	92.9	-	98.73	93.9	98.7	98.4	99.8	**99.9**
grid	70.8	41	94.6	84.0	-	99.60	**100**	97.9	99.8	99.3	**100**
hazelnut	78.5	91.1	92.0	99.3	-	99.98	98.3	**100**	**100**	**100**	**100**
leather	84.2	88	90.9	97.1	-	**100**	**100**	**100**	**100**	**100**	**100**
metal nut	70.0	61.1	94.0	96.1	-	99.26	99.9	**100**	**100**	99.8	**100**
pill	74.3	72.9	86.1	88.8	-	96.82	94.9	96.0	97.2	96.8	**98.2**
screw	74.6	74.7	81.3	96.3	-	91.89	88.7	97.0	**98.9**	94.0	96.2
tile	79.4	87.6	97.8	99.4	-	99.88	94.6	98.9	98.9	**100**	**100**
toothbrush	65.3	61.9	**100**	98.6	-	99.65	99.4	99.7	**100**	96.7	**100**
transistor	79.2	56.7	91.5	91.1	-	95.21	96.1	**100**	**100**	99.3	99.4
wood	83.4	95.3	96.5	**99.8**	-	99.12	99.1	99.0	99.5	99.1	99.3
zipper	74.5	51.7	97.9	95.1	-	98.48	99.9	99.5	99.9	99.9	**100**
average	76.2	71.9	92.1	94.9	97.9	98.26	96.1	99.0	**99.4**	98.7	**99.4**



Experiments: Ablations

- Ablation study on statistics estimation in MVTec and STC datasets.

	μ_i	σ_i^2	STC	MVTec
Model **(3a)**	fixed	fixed	75.9	97.6
Model **(3b)**	estimated	fixed	75.1	97.0
Model **(3c)**	estimated	estimated	74.5	98.1
SANFlow (Ours)	fixed	estimated	**76.1**	**98.7**

- Ablation study on loss functions in MVTec and STC datasets.

	$\mathcal{L}_{\text{NLL}}$	$\mathcal{L}_{\text{BCE}}$	$\mathcal{L}_{\text{KL}}$	STC	MVTec
Model **(1a)**	Eq.(2)	✗	✗	72.6	98.3
Model **(1b)**	Eq.(3)&(5)	✗	✗	73.5	98.2
Model **(1c)**	Eq.(3)&(5)	✓	✗	73.1	98.3
Model **(1d)**	Eq.(3)&(5)	✗	✓	74.2	98.5
SANFlow	Eq.(3)&(5)	✓	✓	**76.1**	**98.7**

- Ablation study on anomaly augmentation in MVTec and STC datasets.

	Augmentation	$\mathcal{L}_{\text{NLL}}$	$\mathcal{L}_{\text{BCE}}$	$\mathcal{L}_{\text{KL}}$	STC	MVTec
Model **(1d)**	✓	✓	✗	✓	74.2	98.5
Model **(2a)**	✗	✓	✗	✓	73.6	98.4
SANFlow	✓	✓	✓	✓	**76.1**	**98.7**