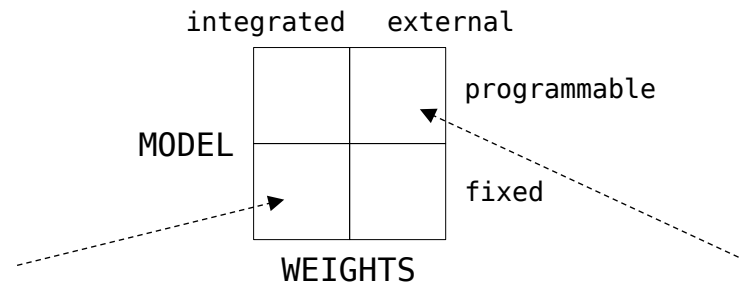


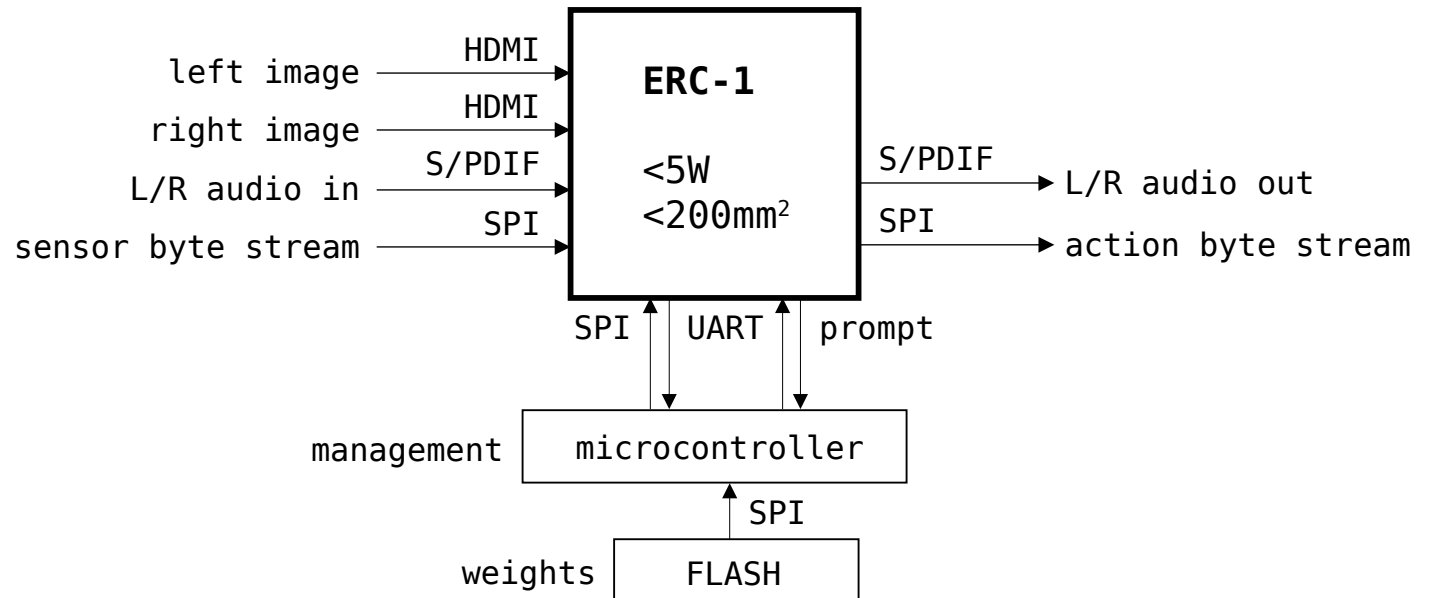
hardened components
 integrated SRAM weights
 fixed model
 no software, weights only



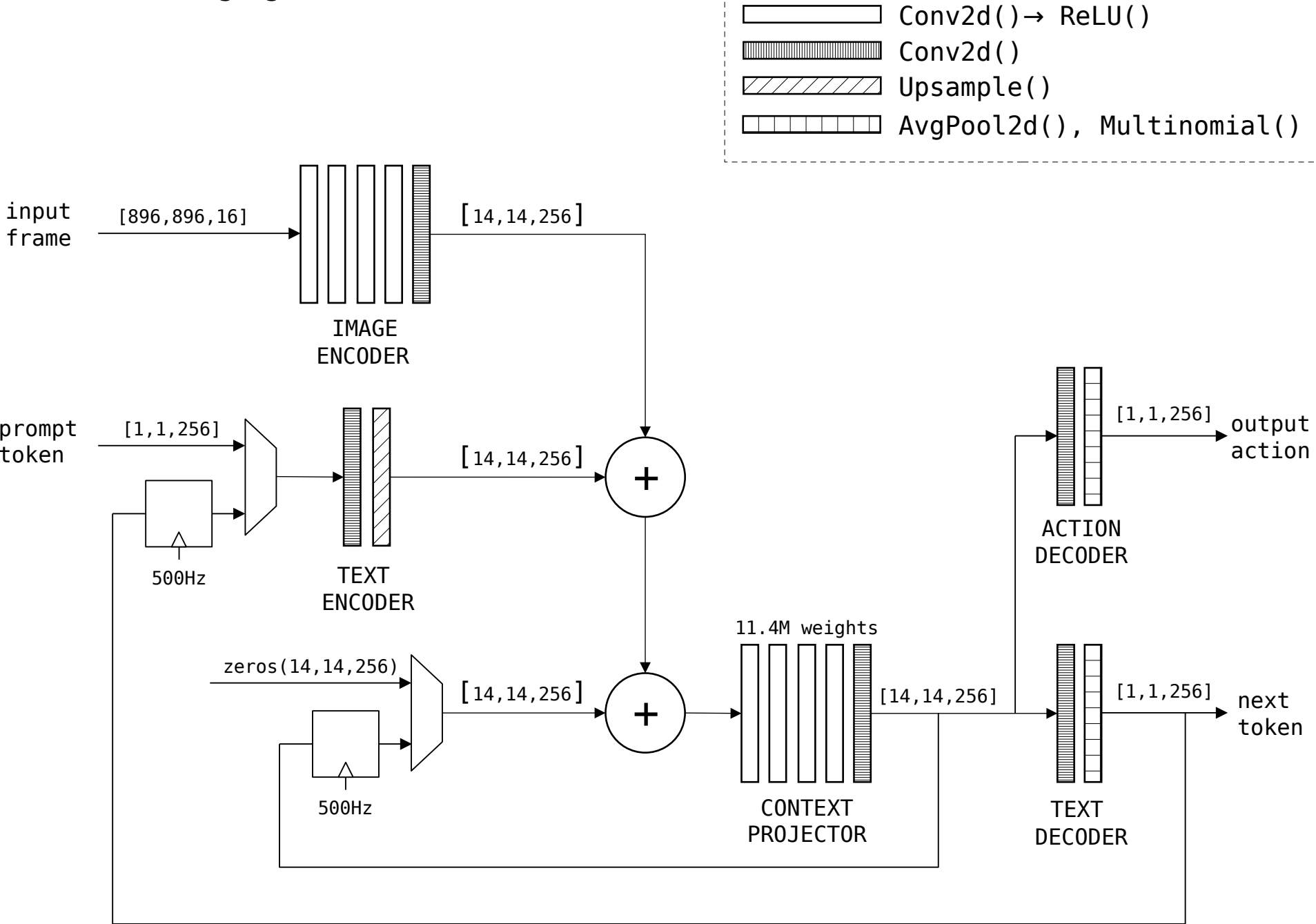
AI accelerator
 external DRAM weights
 programmable model
 Pytorch software stack

Embedded Robot Controller

Autonomous robots
 Interactive signage
 Nuclear fusion reactors



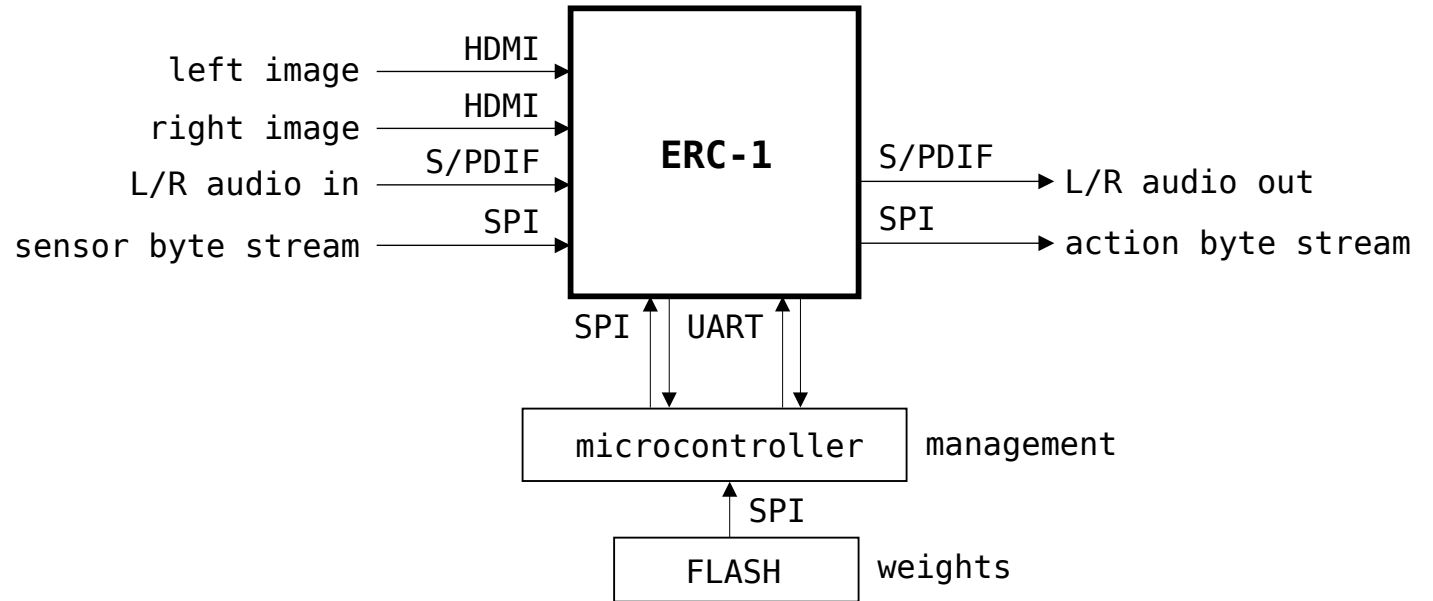
CNN Vision-Language-Action Model



BACKUP

Embedded Robot Controller

Autonomous robots
Interactive signage
Nuclear fusion reactors



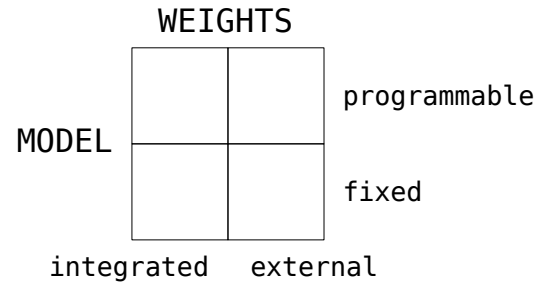
hardened components

integrated SRAM/ROM weights
fixed model

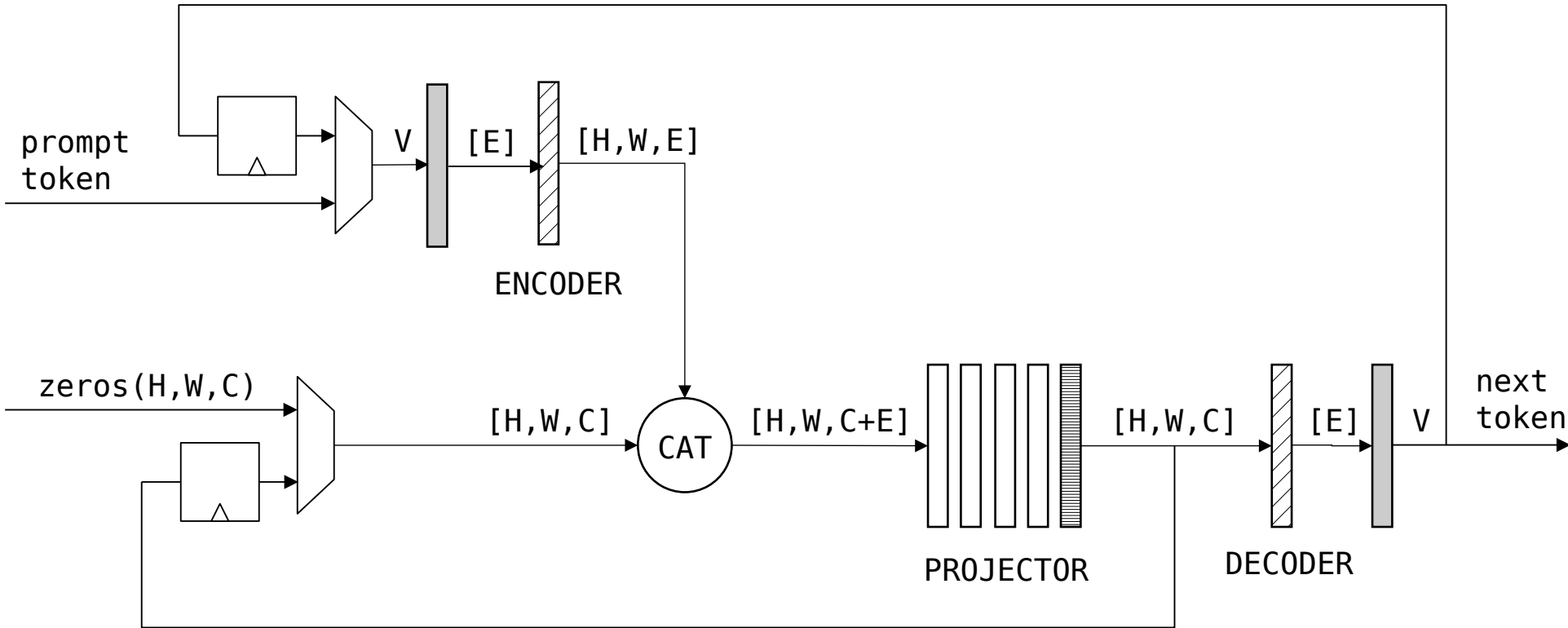






AI accelerator

external DRAM weights
programmable model



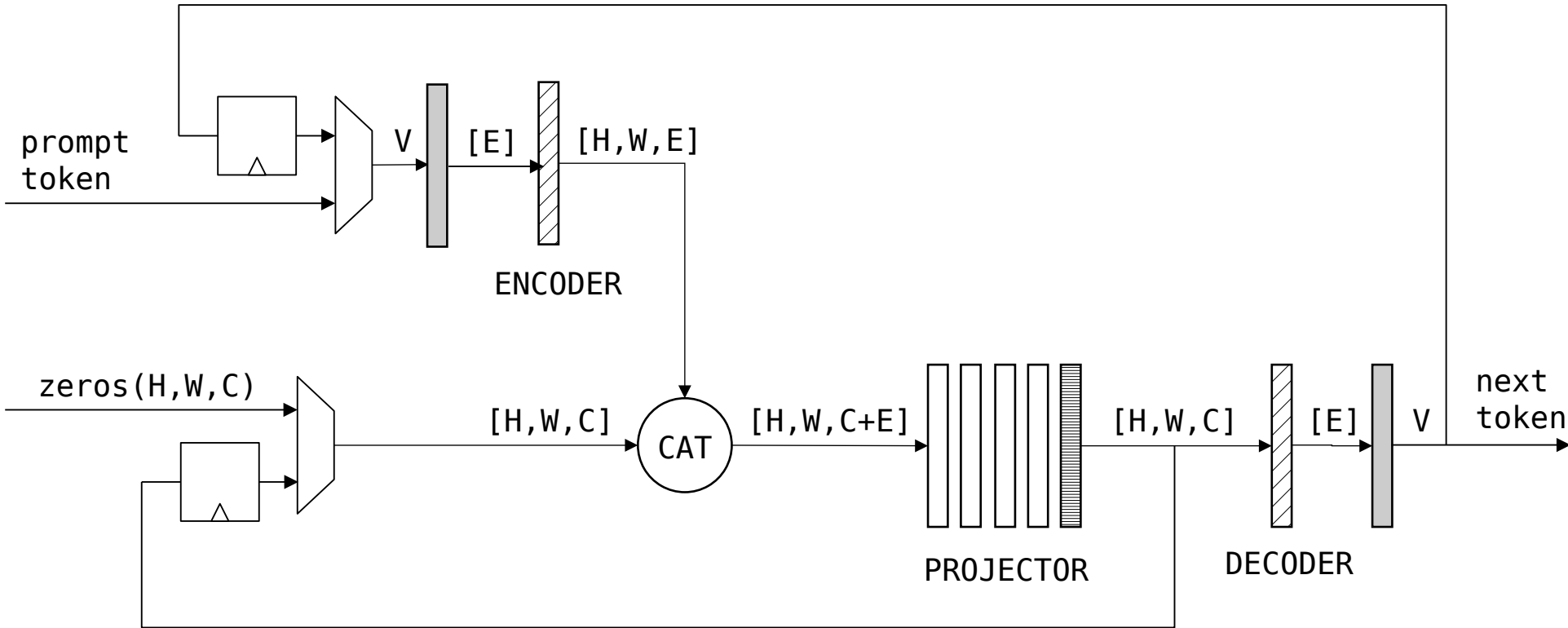
CNN Language Model







-  **Conv2d() → ReLU()**
-  **Conv2d()**
-  **Upsample(), AvgPool2d()**
-  **FROZEN**

V	50257
E	256
C	512
H,W	8,8

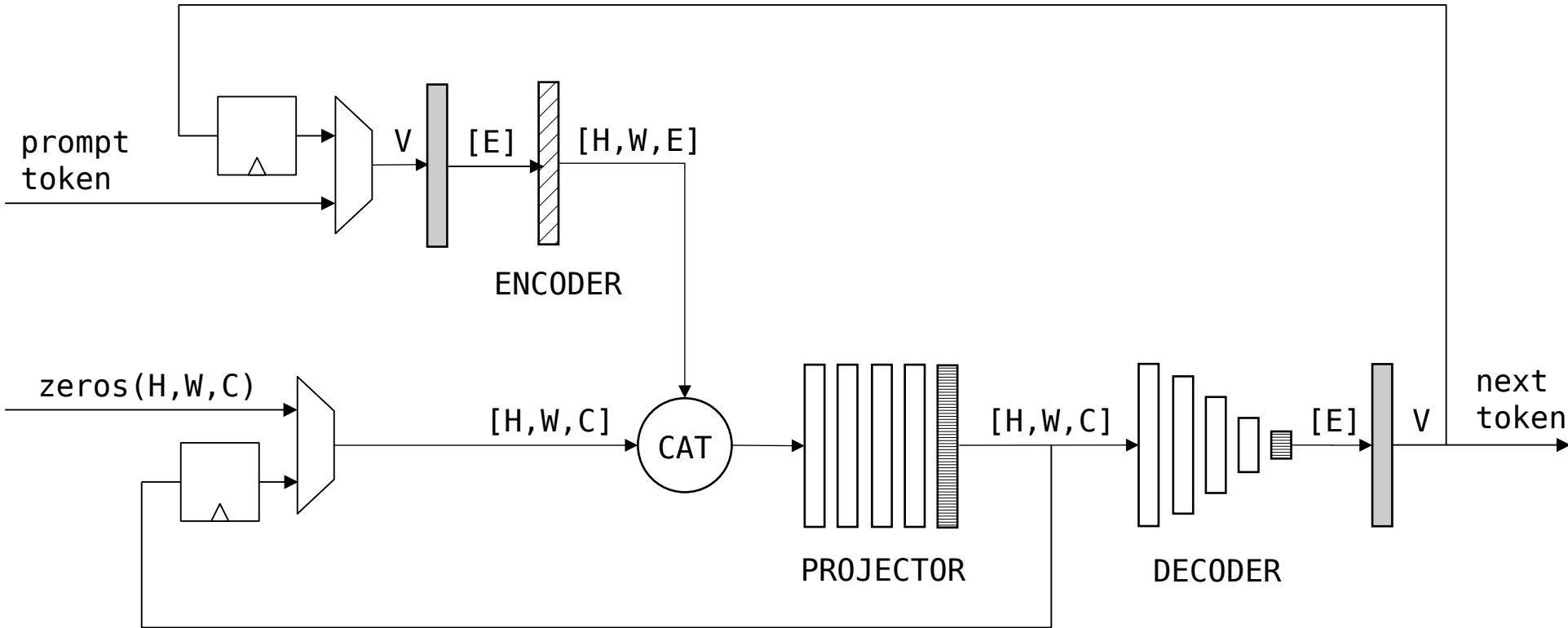
CNN Language Model







-  **Conv2d() → ReLU()**
-  **Conv2d()**
-  **Upsample(), AvgPool2d()**
-  **FROZEN**

V	50257
E	256
C	512
H,W	8,8

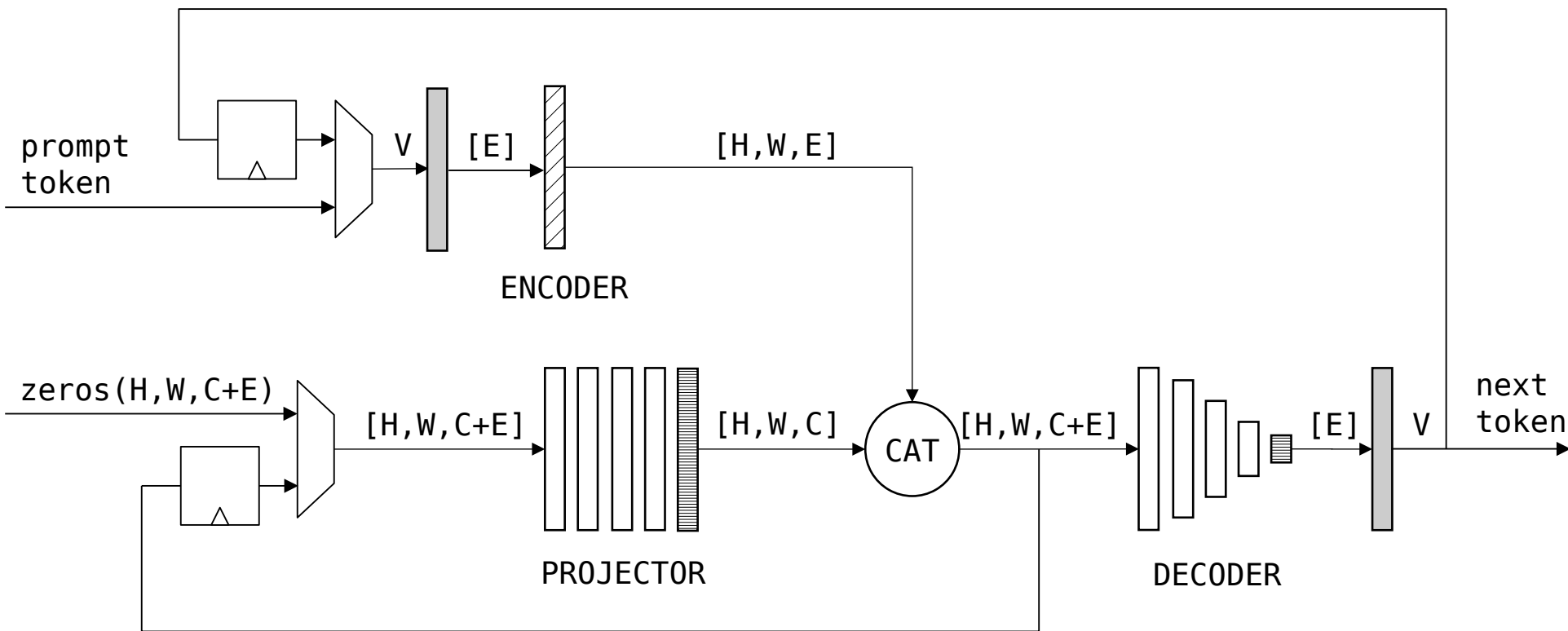
CNN Language Model

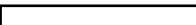





-  Conv2d() \rightarrow ReLU()
-  Conv2d()
-  Upsample()
-  FROZEN

E	256
V	50257
C	256
H,W	14,14

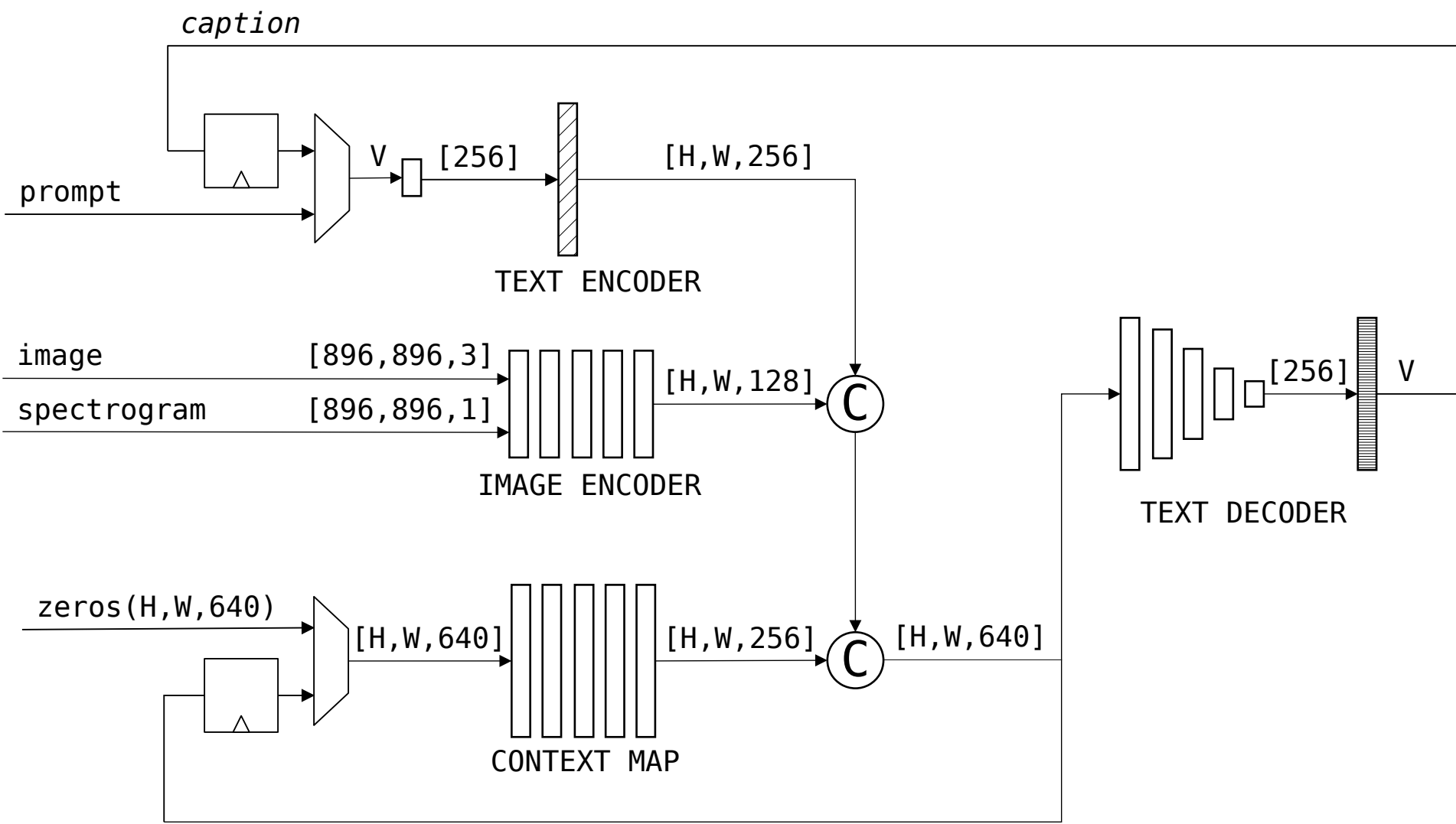
CNN Language Model






-  Conv2d() → ReLU()
-  Conv2d()
-  Upsample()
-  FROZEN

E	256
V	50257
C	256
H,W	14, 14

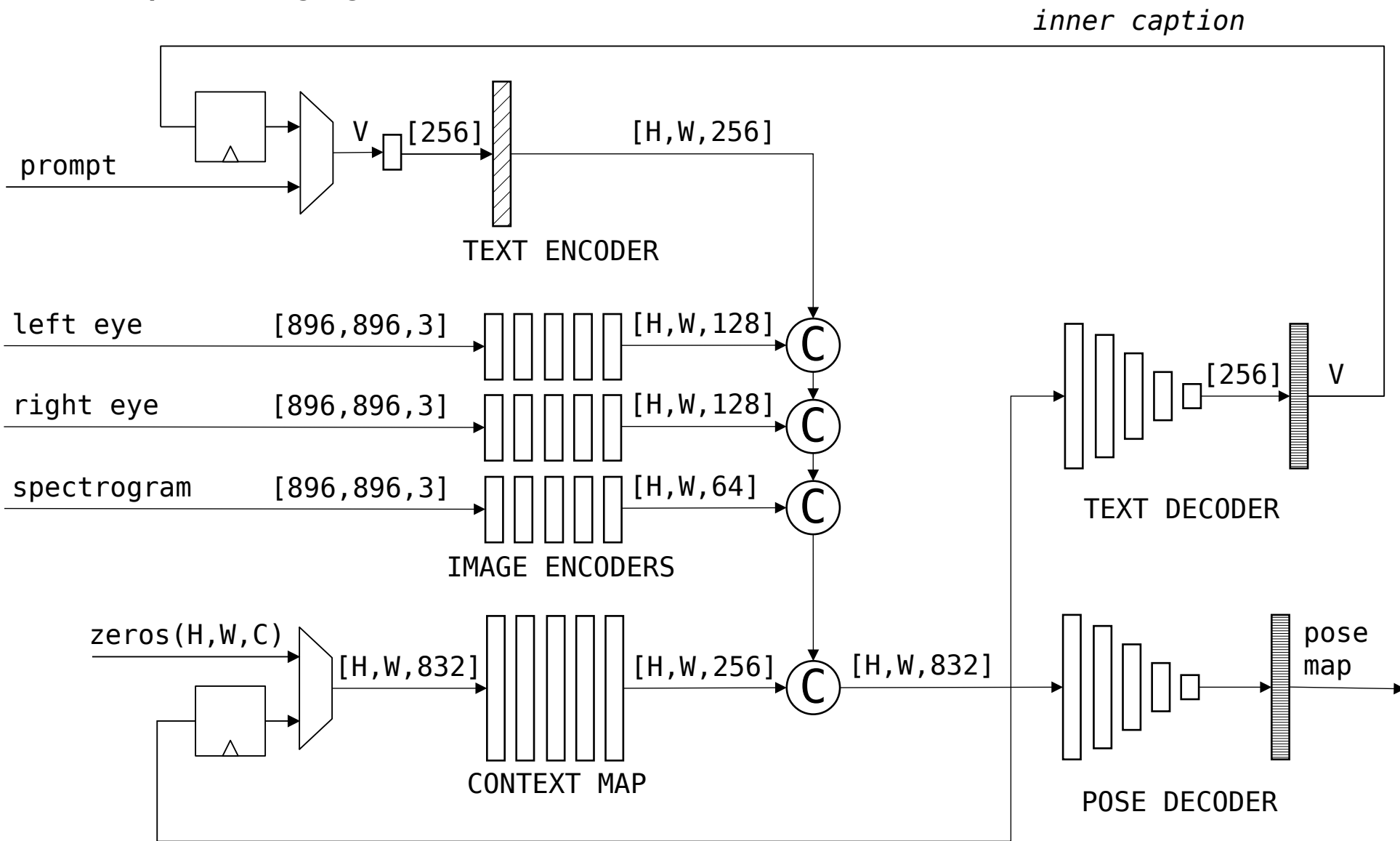
CNN Vision-Language Model

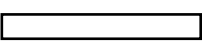

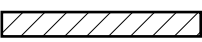


-  Conv2d() → ReLU()
-  Conv2d()
-  Upsample() → Conv2d()

H,W 32,32

CNN Perception-Language-Action Model



-  Conv2d() \rightarrow BatchNorm2d() \rightarrow ReLU()
-  Conv2d()
-  Upsample() \rightarrow Conv2d()

H,W 32,32

PHASE 1:

CNN Perception-Language-Action model (CNN-PLA), running on GPU
>20fps, <100ms latency, <400W

PHASE 2:

CNN-PLA model running on FPGA
>100fps, <15ms latency, <100W

PHASE 3:

CNN-PLA model running on ASIC at >300fps, <5ms latency, <10W

Seated humanoid :

1-DoF spine

2-DoF shoulder

2-DoF neck

6-DoF arms

6-DoF hands

Data collection :

VR teleoperated

spine/shoulder : swivel chair with up/down control

neck : VR gyroscope

arms/hands : TBD

ROBOT

896x896 RGB at 120Hz, left and right eye

896x896 spectrogram at 120Hz, audio and tactile signals

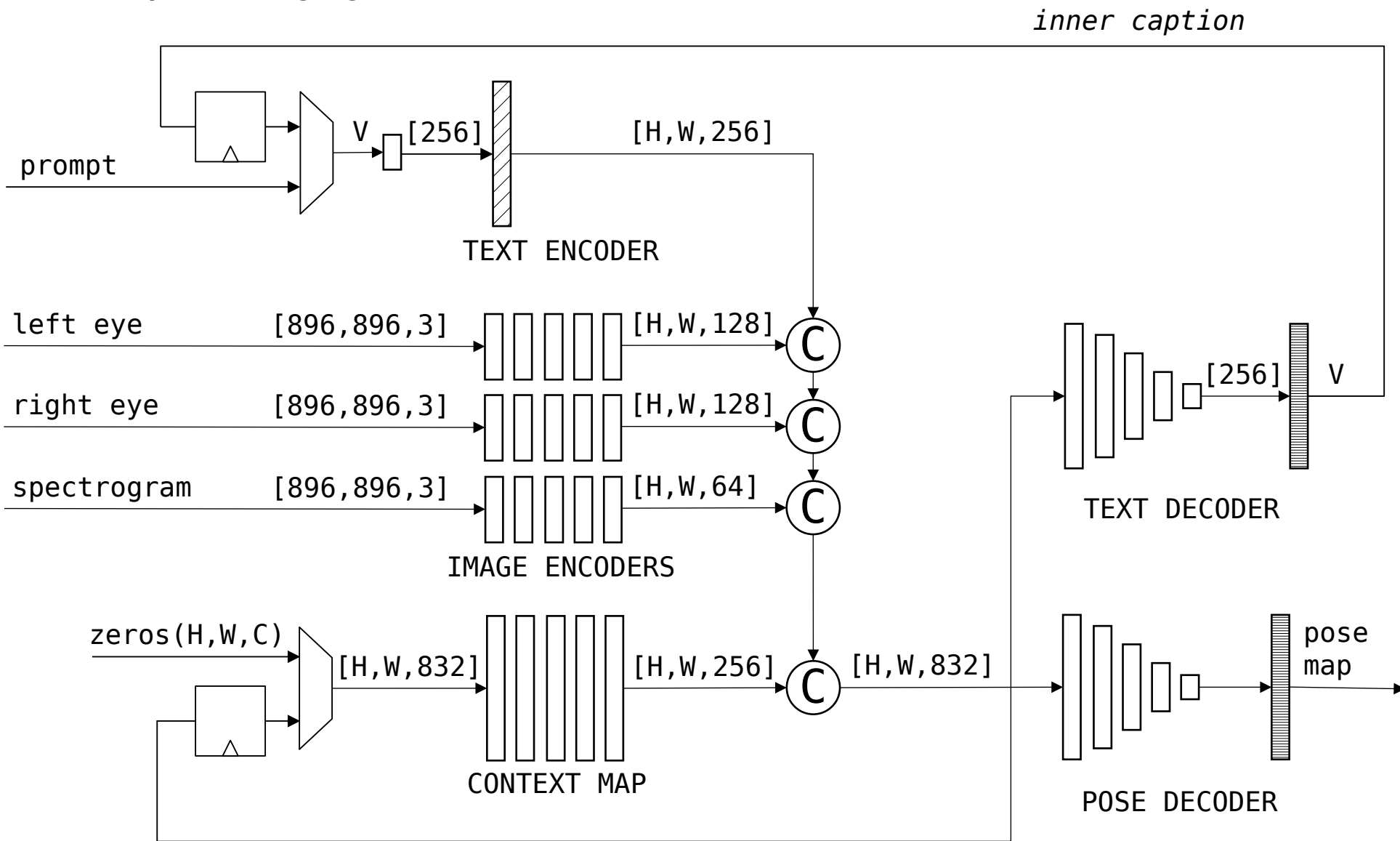
29-DoF target pose at 120Hz

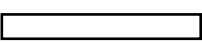

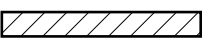
TELEOPERATOR

transcribed verbal commentary at 120 characters/s

synthetically captioned using open source VL model, 1 character/frame

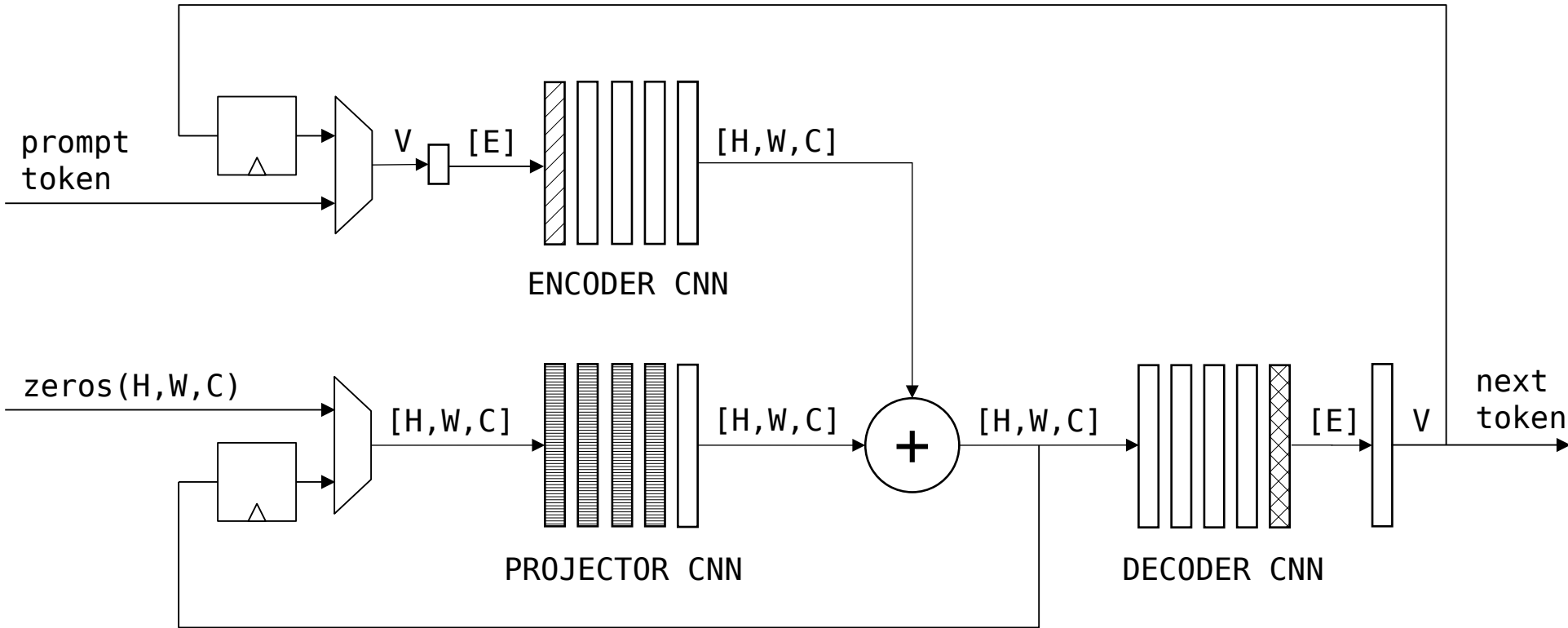
CNN Perception-Language-Action Model



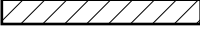



-  Conv2d() \rightarrow BatchNorm2d() \rightarrow ReLU()
-  Conv2d()
-  Upsample() \rightarrow Conv2d()

H,W 32,32

CNN Language Model



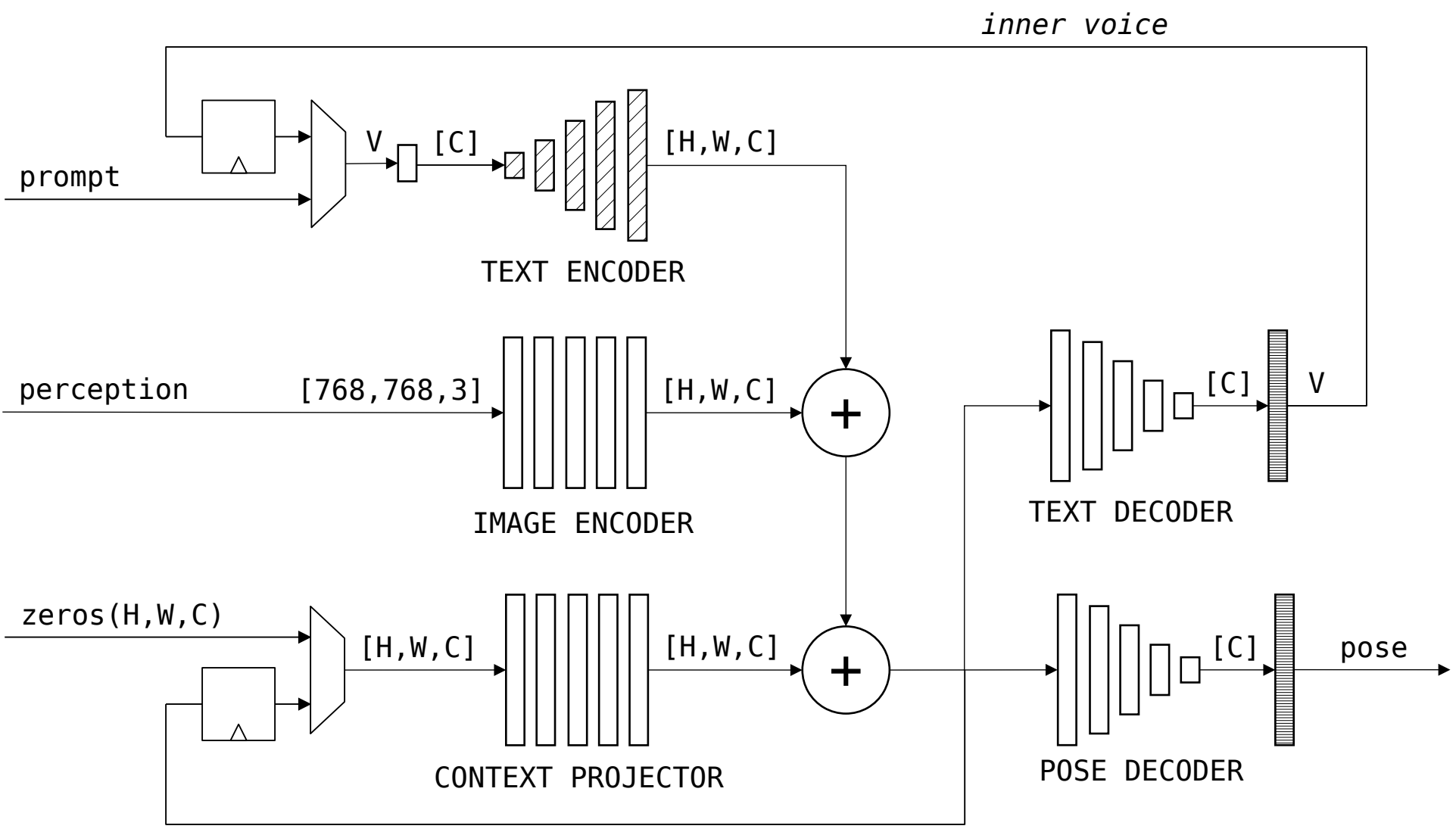
-  Conv2d() \rightarrow ReLU()
-  Conv2d()
-  Upsample()
-  AvgPool()

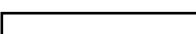

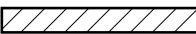
E	256
V	50257
C	384
H,W	14,14

Devices

- TE256A : Token Encoder, vocab_size=256, ASCII
- TD256A : Token Decoder, vocab_size=256, ASCII
- CP384V : Context Projector, n_embd=384, VGG
- IE384R : Image Encoder, n_embd=384, Resnet
- TD
-

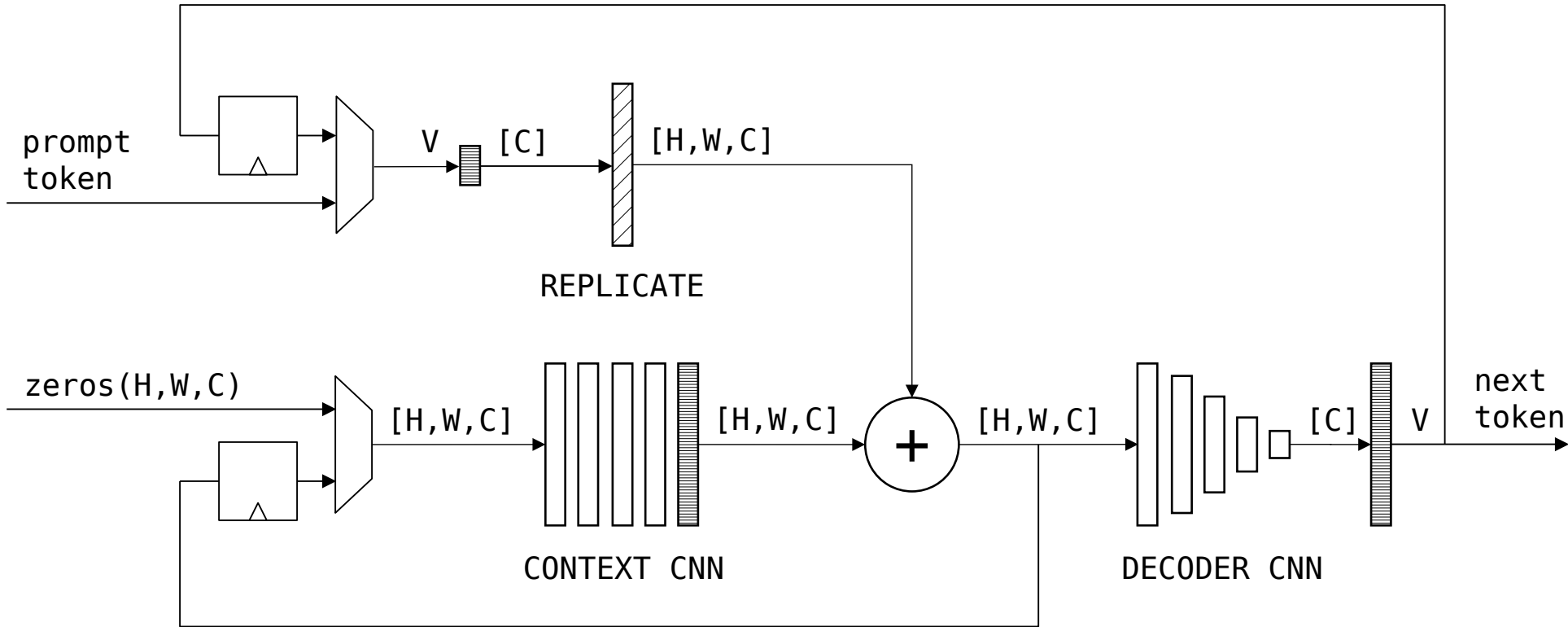
alt = lite-base-resnet-xeno

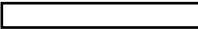




-  Conv2d(kernel=3)→ ReLU()
-  Conv2d(kernel=1)
-  Upsample(scale=3)→ Conv2d(kernel=3)→ ReLU()

V	256
C	384
H,W	81,81

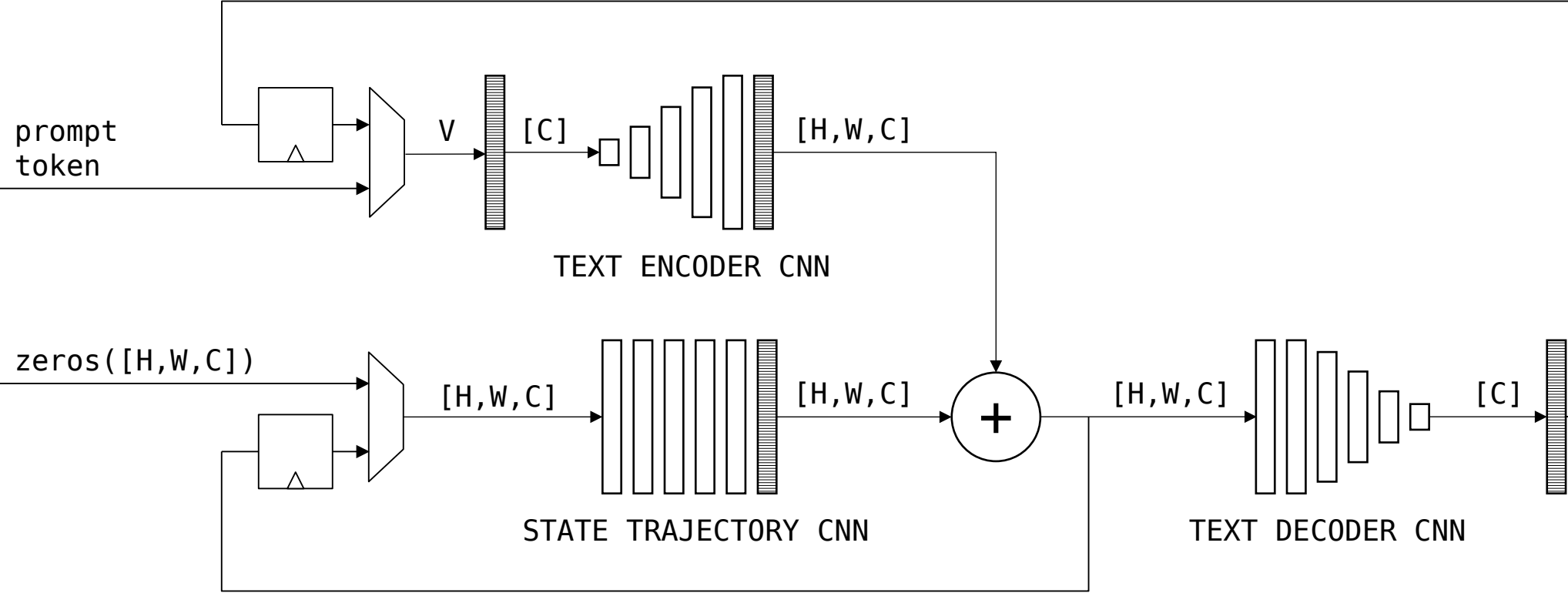
alt2



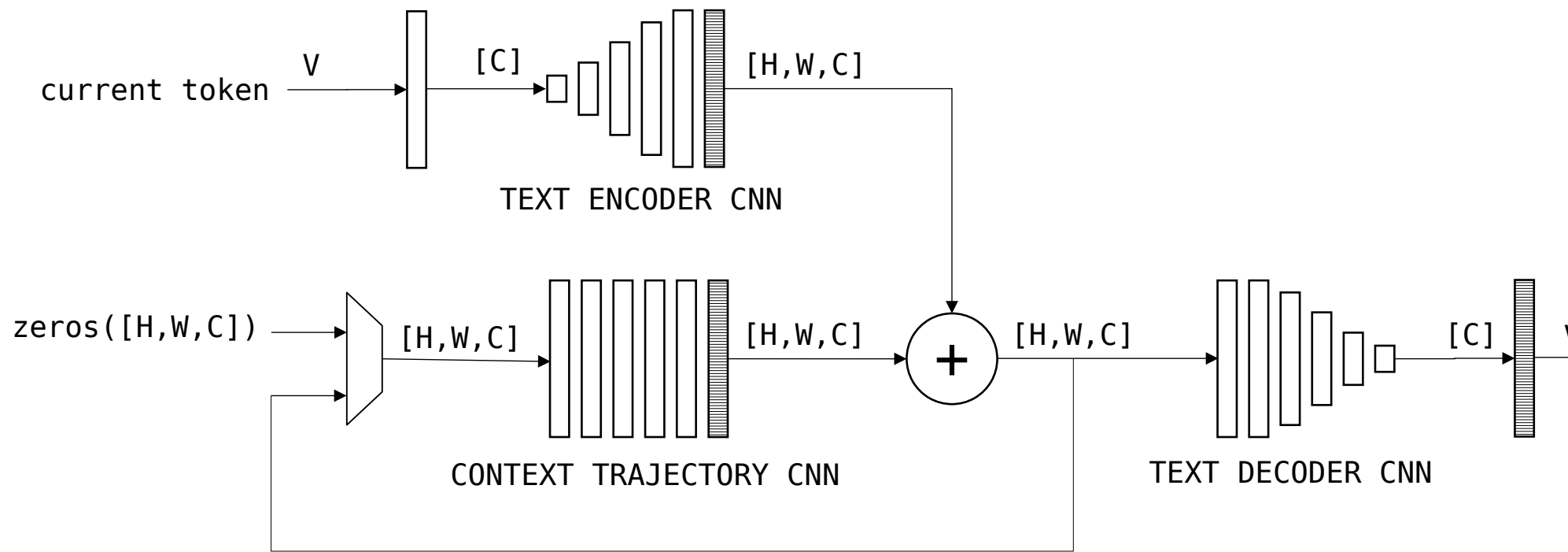
 Conv2d(kernel=3)→ BatchNorm2d()→ ReLU()
 Conv2d(kernel=1)
 Upsample()

V	256
C	384
H,W	81,81

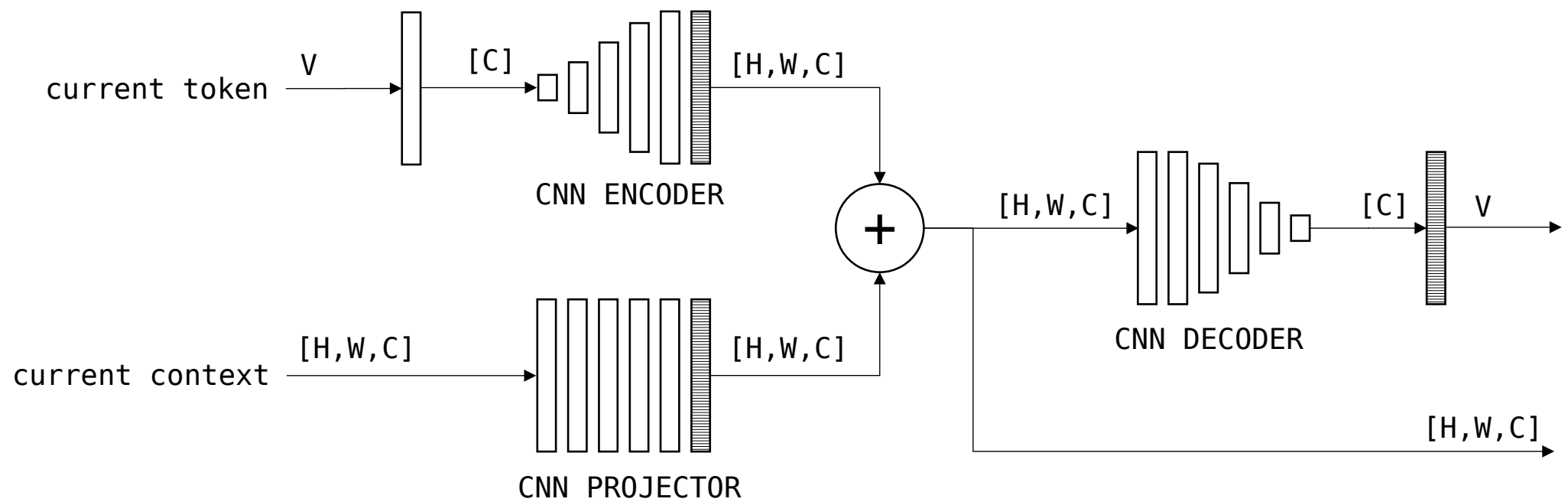
alt1



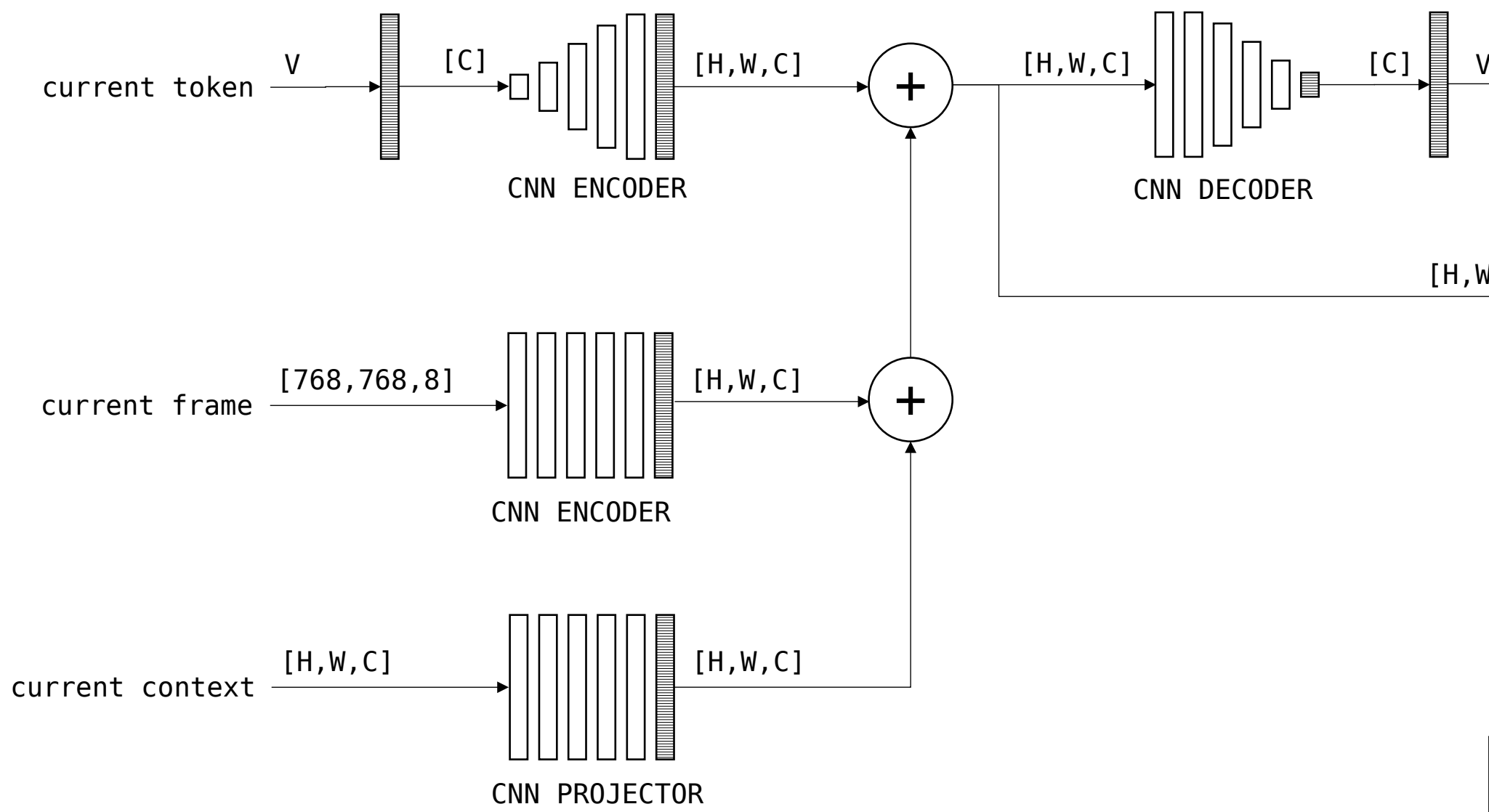
alt1



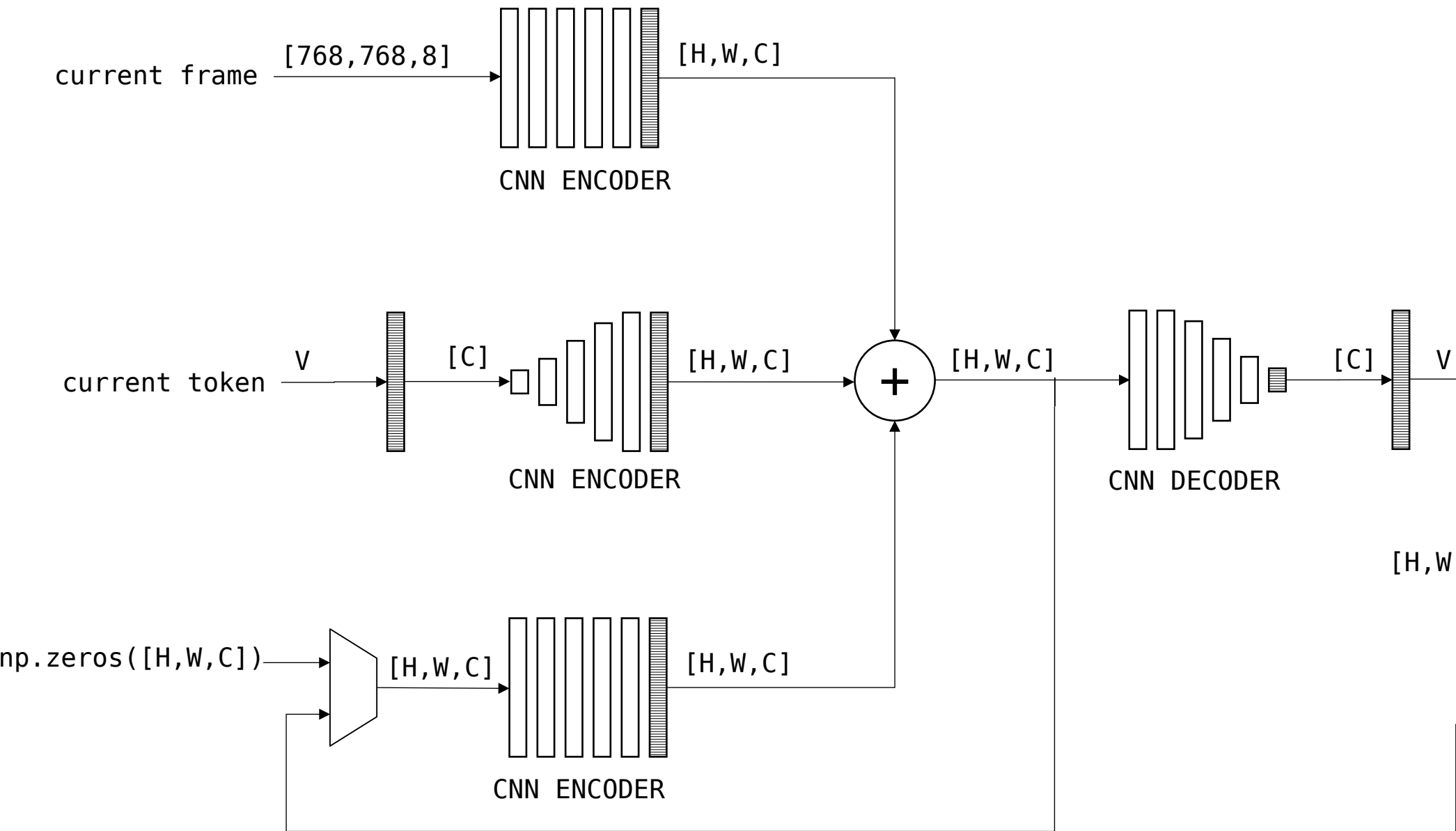
alt1



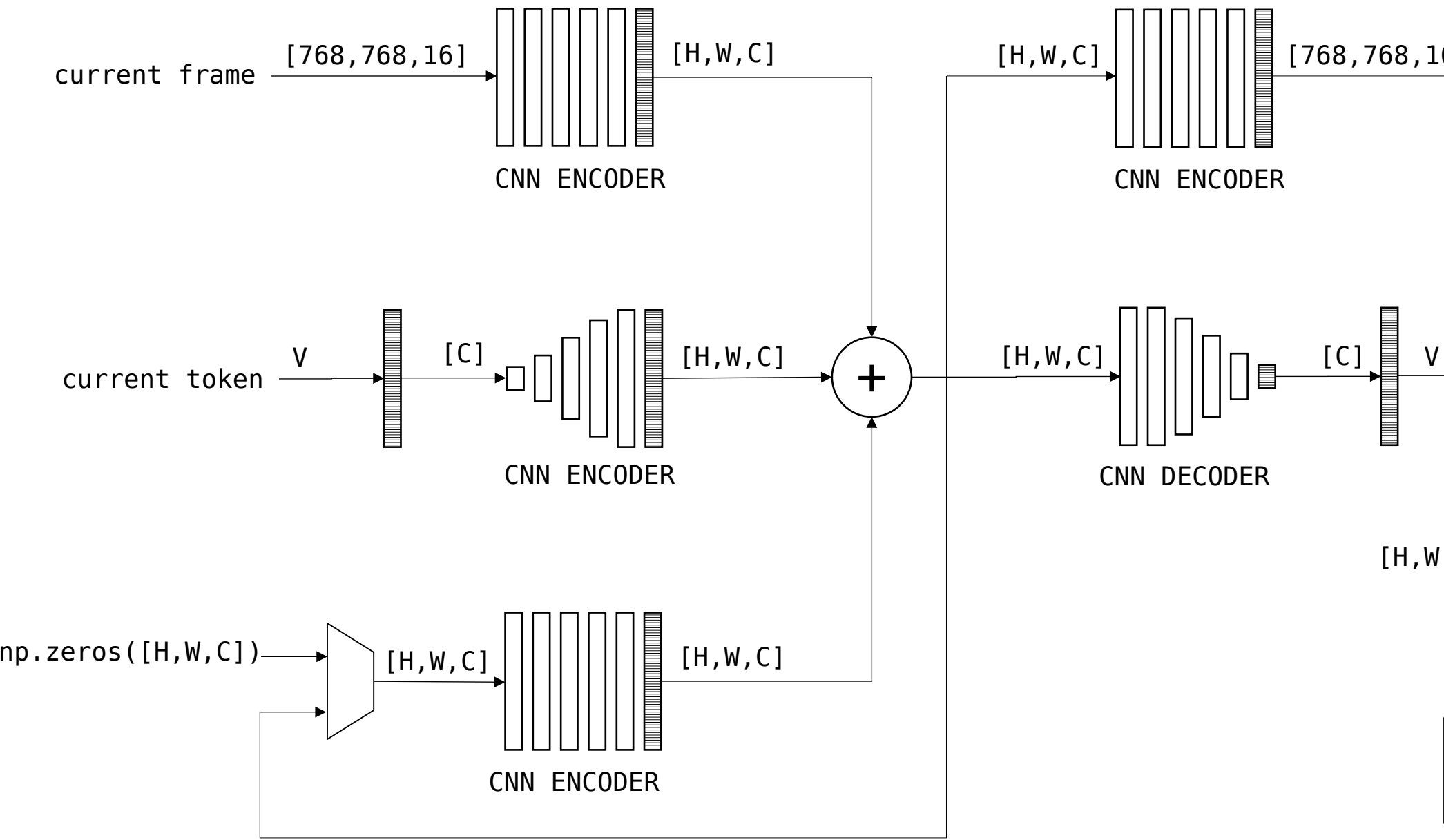
multimodal alt1



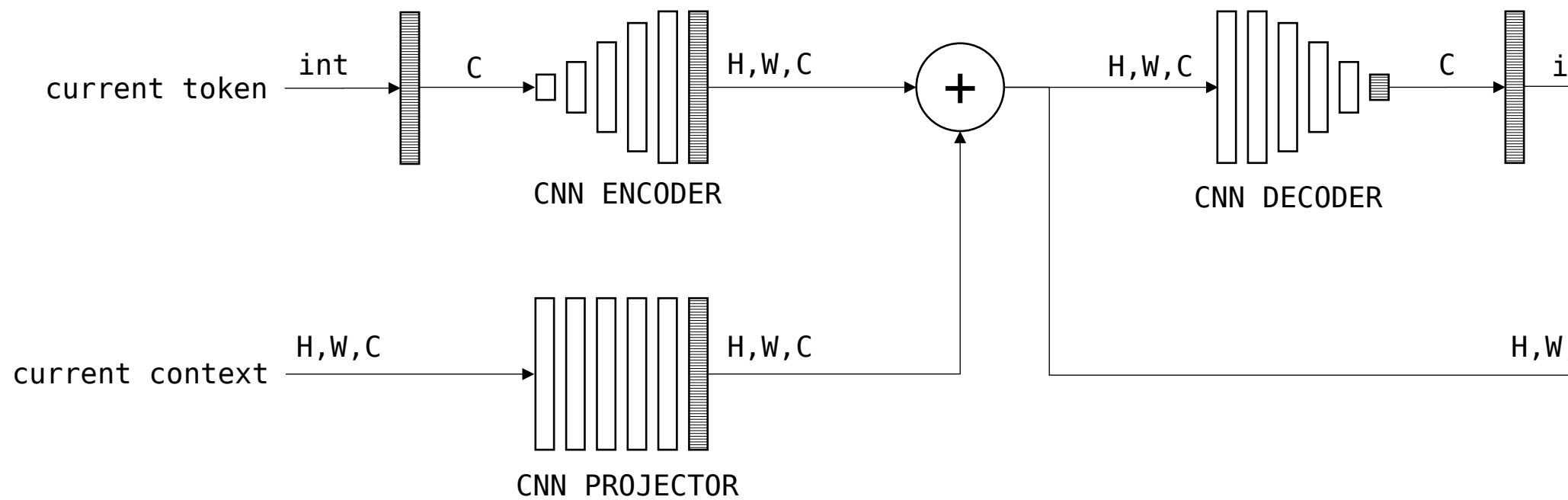
multimodal alt1

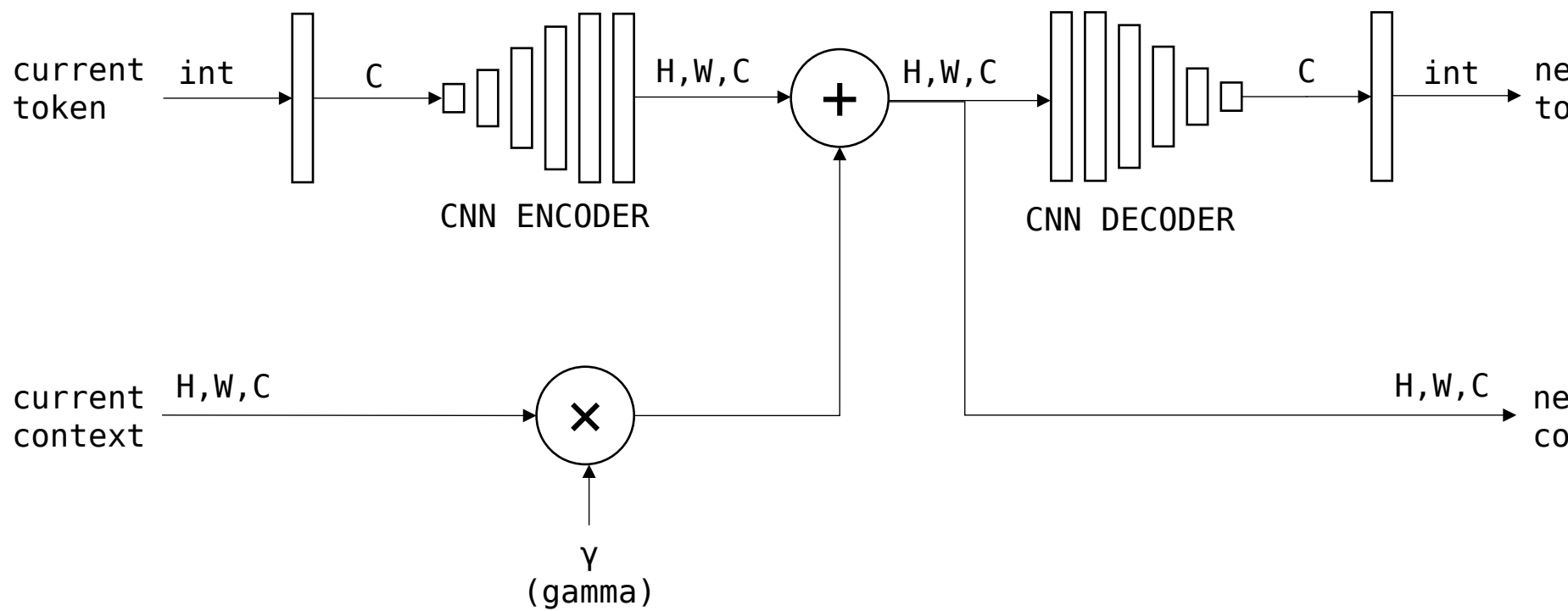


multimodal alt1

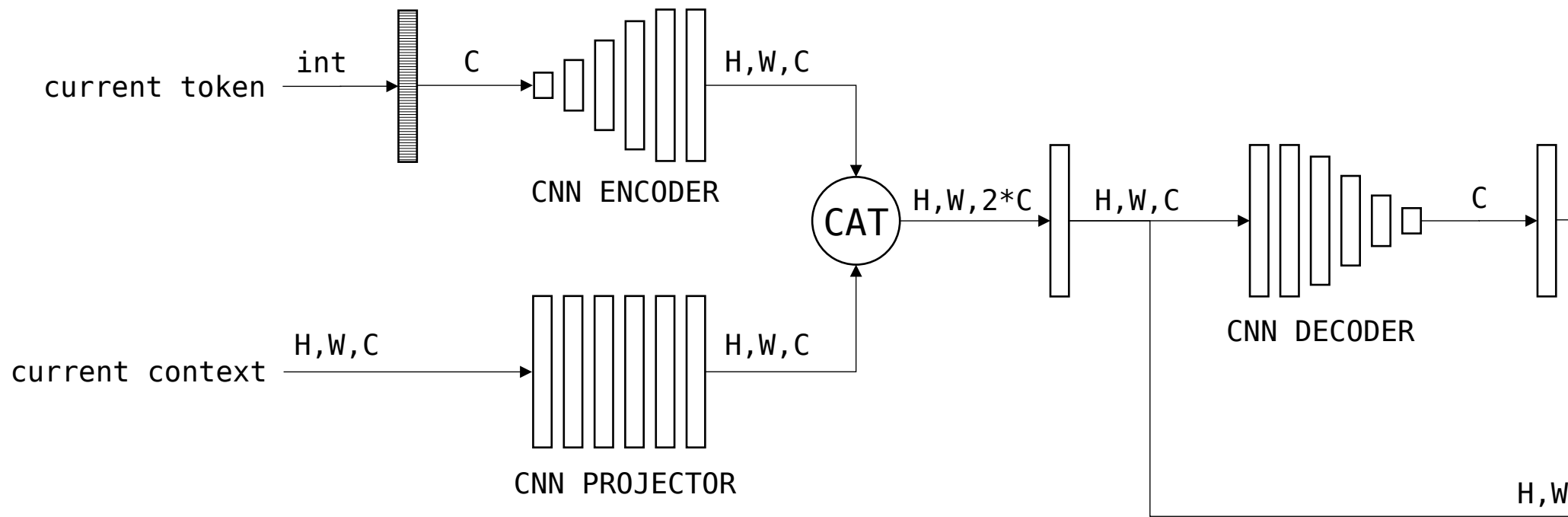


alt1





alt2



alt3

