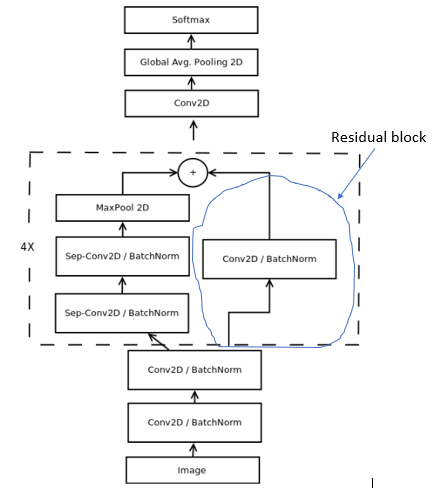
# Ý tưởng thuật toán

Bao gồm 2 bước chính:

* Phát hiện vị trí khuôn mặt trong bức ảnh: sử dụng đặc trưng HOG + svm để tìm vị trí khuôn mặt trong bức ảnh.
* Nhận diện cảm xúc khuôn mặt: đưa vị trí khuôn mặt đã được trích xuất vào mô hình mạng Mini\_Xception được training trên tập dữ liệu FER của Kaggle để nhận diện cảm xúc. Các cảm xúc bao gồm: 0=Angry, 1=Disgust, 2=Fear, 3=Happy, 4=Sad, 5=Surprise, 6=Neutral.

# Mini\_Xception model

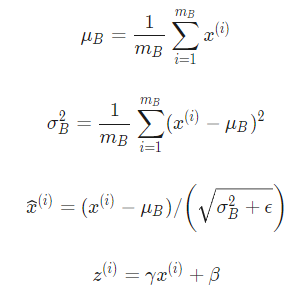


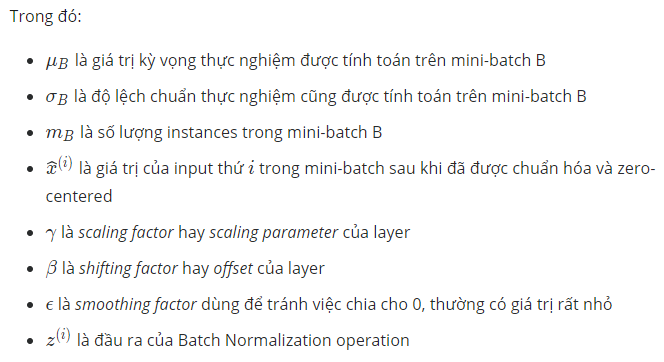
Phần này phải vừa đọc code vừa xem hình mới hiểu được

## Batch Normalization

Ý tưởng của Batch Normalization là thêm một công đoạn (operation) ngay trước activation function cho mỗi layer. Operation đó có nhiệm vụ chuẩn hóa (normalizing) và zero-centering (mean subtracting) các inputs (mean của inputs sẽ là 0). Từ đó các hệ số trở nên cân bằng hơn không quá nhỏ hoặc quá lớn nên sẽ giúp model dễ hội tụ hơn. Để thực hiện normalizing và zero-centering, Batch-Norm sẽ tính độ lệch chuẩn và phương sai của các inputs trên các mini-batches, sau đó sử dụng hai parameter là γ và β để thực hiện việc scaling và shifting.

Batch normalization algorithm:





Trong quá trình testing, chúng ta sẽ không có các mini-batches để tính toán phương sai và độ lệch chuẩn. Do vậy chúng ta sẽ sử dụng phương sai và độ lệch chuẩn của training set. Hai giá trị này thường được tính toán sử dụng EWMA - Exponentially Weighted Averages. Chung quy lại, mỗi Batch-Norm layer sẽ có 4 parameter cần phải học: γ (scale), β (shift), μ (mean) và σ (standard deviation).

Một số lợi ích của Batch Normalization:

* Giảm thiểu đến mức tối thiểu hiện tượng Vanishing / Exploding gradients và chúng ta có thể quay lại sử dụng các activation function như sigmoid hay tanh.
* Giảm thiểu sự phụ thuộc vào quá trình weight initialization
* Có thể sử dụng learning rate lớn hơn để tăng tốc quá trình training
* Batch-Norm có thể được sử dụng như một regularizer giúp giảm overfitting

## Residual block

Ý tưởng của residual block là feed foward x(input) qua một số layer conv-max-conv, ta thu được F(x) sau đó cộng thêm x vào H(x) = F(x) + x . Model sẽ dễ học hơn khi chúng ta thêm feature từ layer trước vào.

