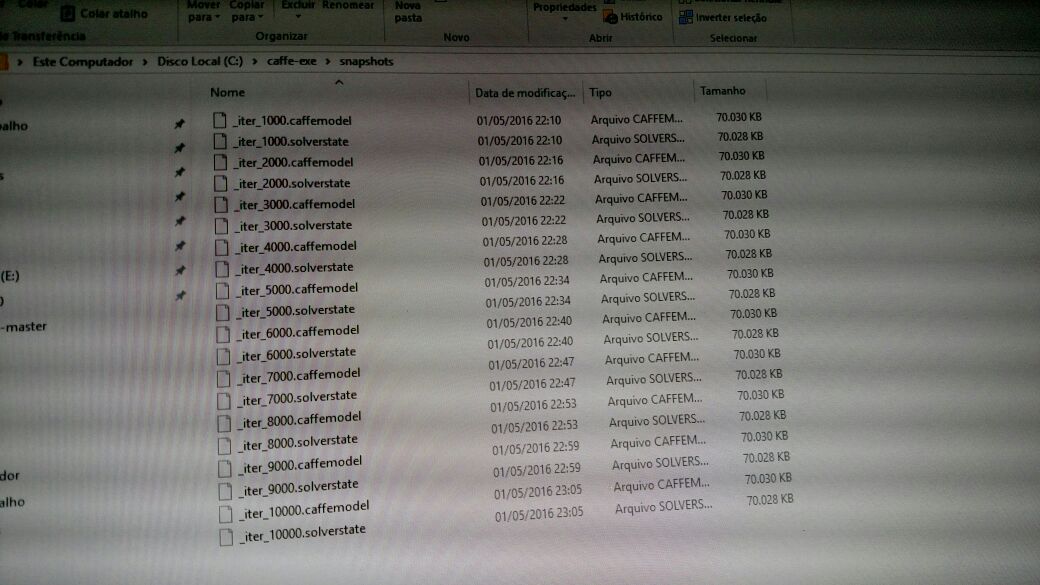
Anexo 01 – Cópias de Telas

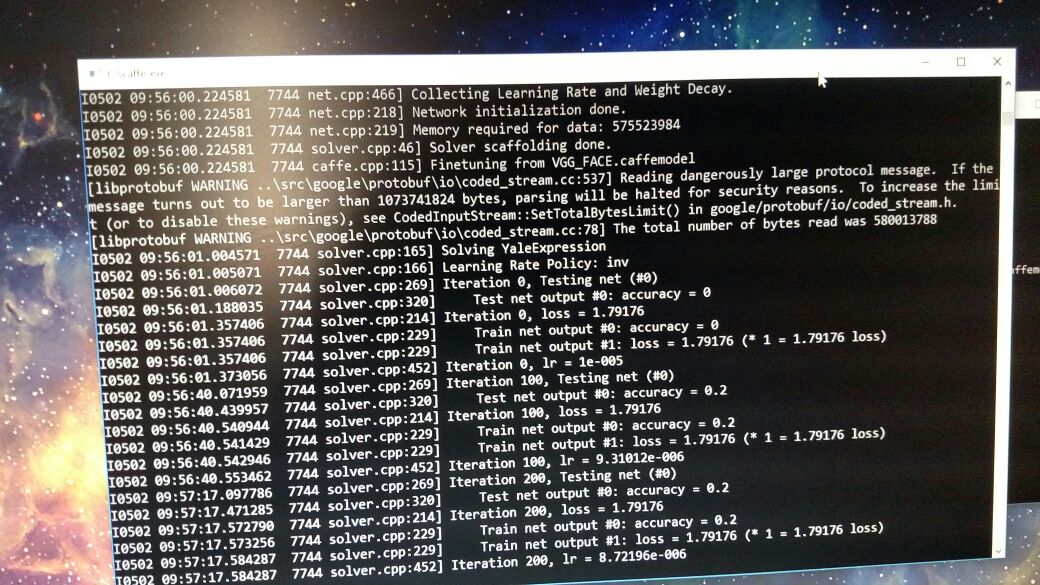
**Data :** 01/05/2016 **(Ambiente Windows) : Início:** 22:10 **- Término:** 23:05

**Resultado :** 10.000 iterações, programadas no solver, terminaram sem que se conseguisse gravar a tela com para se verificar a acurácia obtida (ver gravação dos arquivos snapshots a seguir).



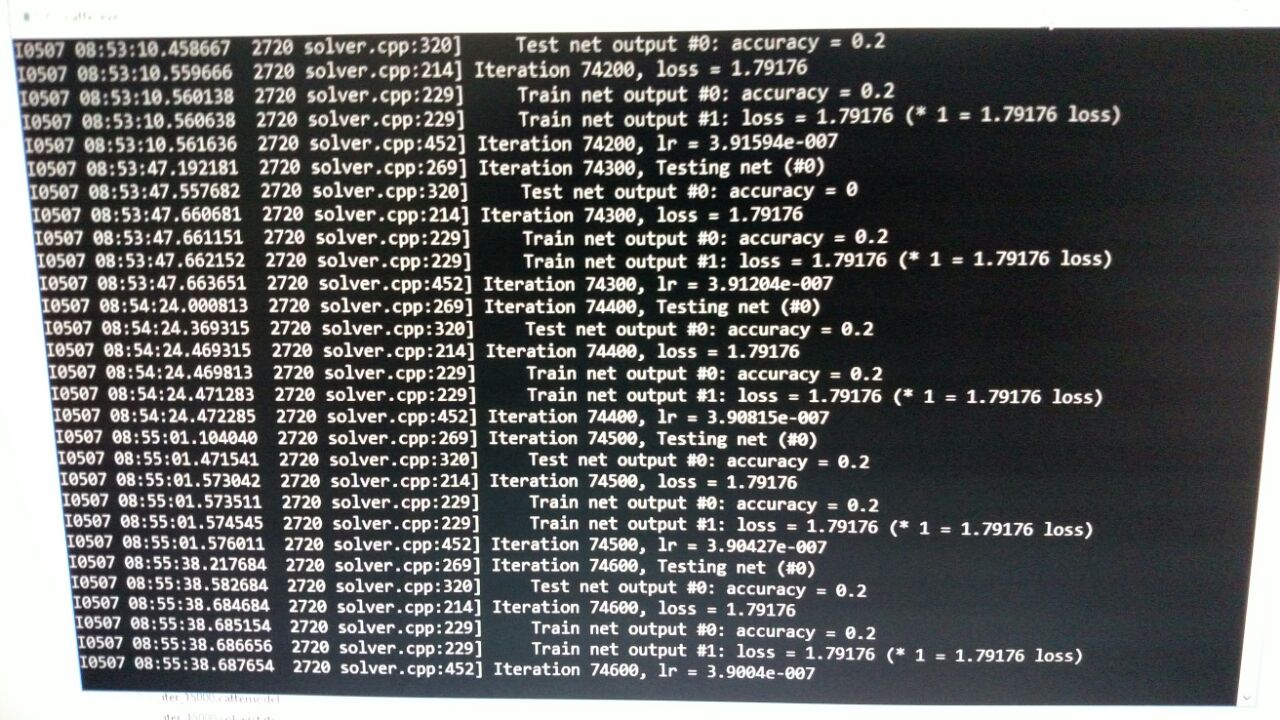
**Data :** 02/05/2016 **(Ambiente Windows) : Início:** 9:50 **- Término:** 10:10

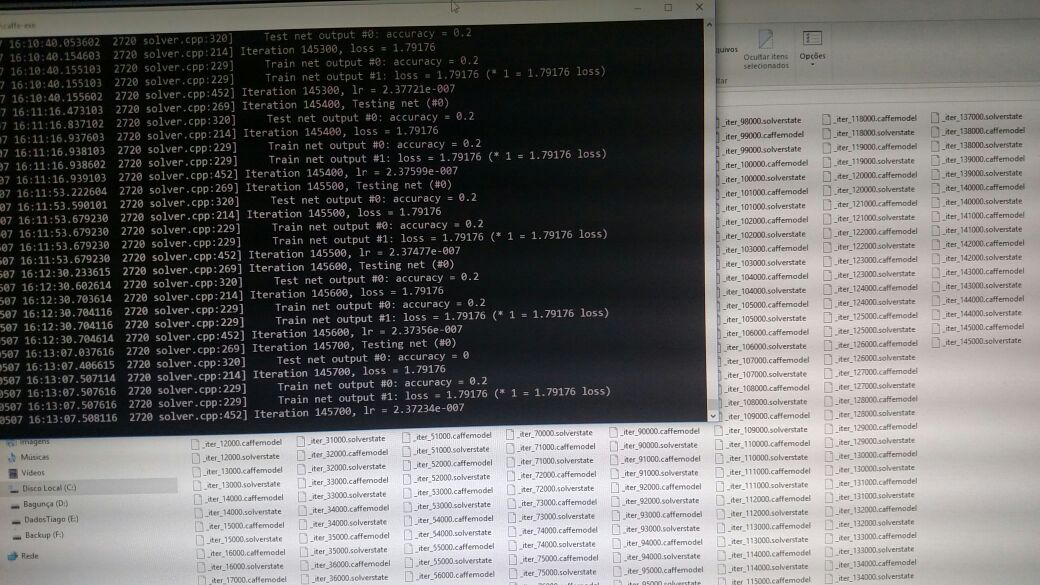
**Resultado :** No início a rede apresentou uma animadora acurácia de treino e de teste de 20%, (ver imagem a seguir) porém a rede terminou de processar as 10.000 iterações que estavam programadas sem que tais acurácias evoluíssem (ver de tela com displays da rede com a acurácia de 20%).

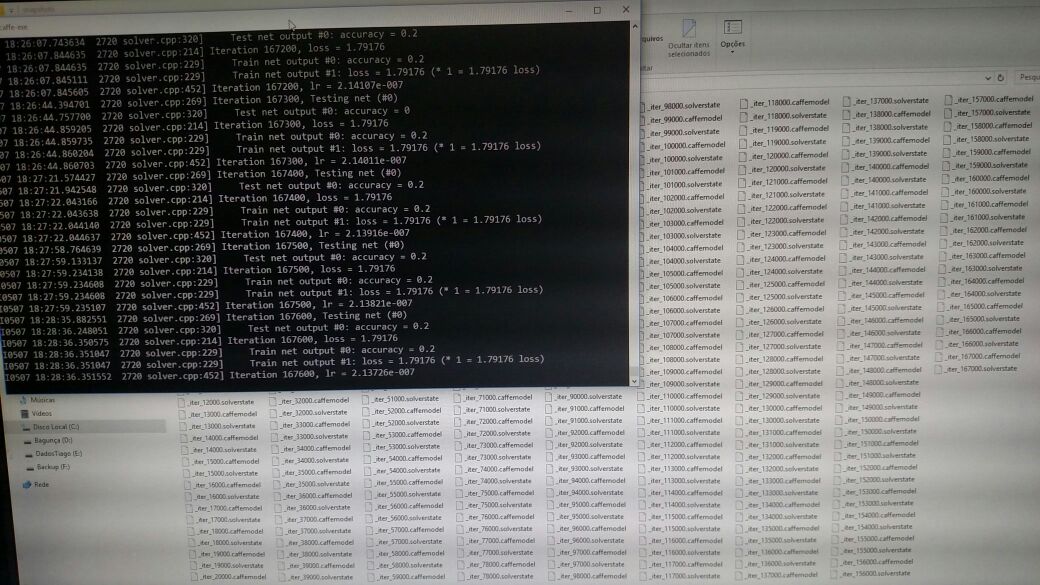
.

**Data :** 07/05/2016 **(Ambiente Windows) – Vários Processamentos durante o dia**

**Resultado :** Entendendo que o número de iterações máximas podia ser um limitante para a melhoria da acurácia , o número máximo de iterações foi aumentado em três novos processamentos da rede; 1º) para de 10.000 para 100.000, 2º) de 100.000 para 150.000 e, 3º) De 150.000 para 200.000) entretanto, após cada mudança a acurácia manteve-se em 20%(ver de cópias de telas com as acurácias para: com 76.400 iterações para um número máximo de iterações 100.000, com 145.000 iterações para um número máximo de iterações de 150.000 e 167.500 iterações para um número máximo de iterações de 200.000).

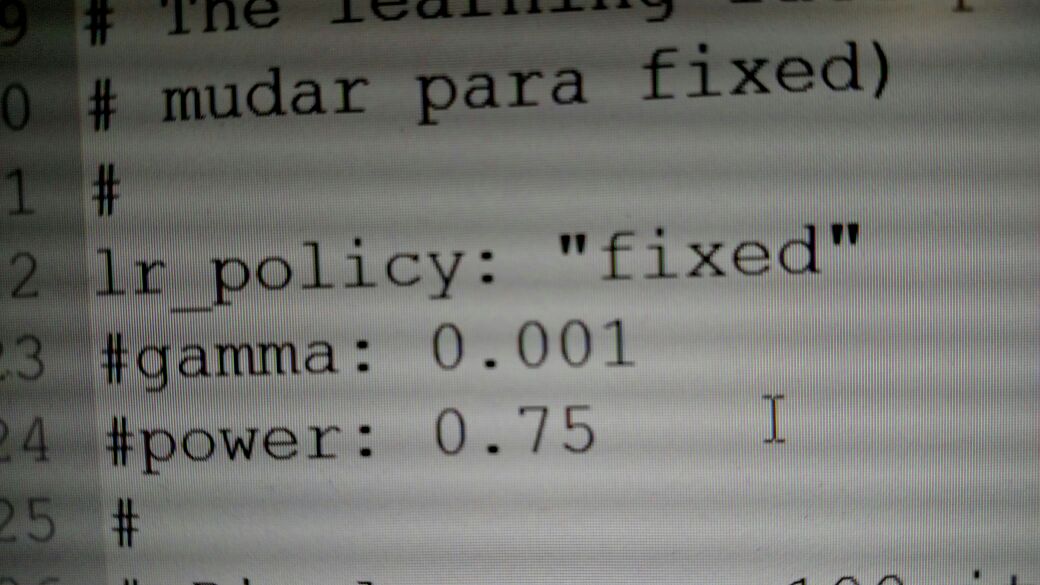


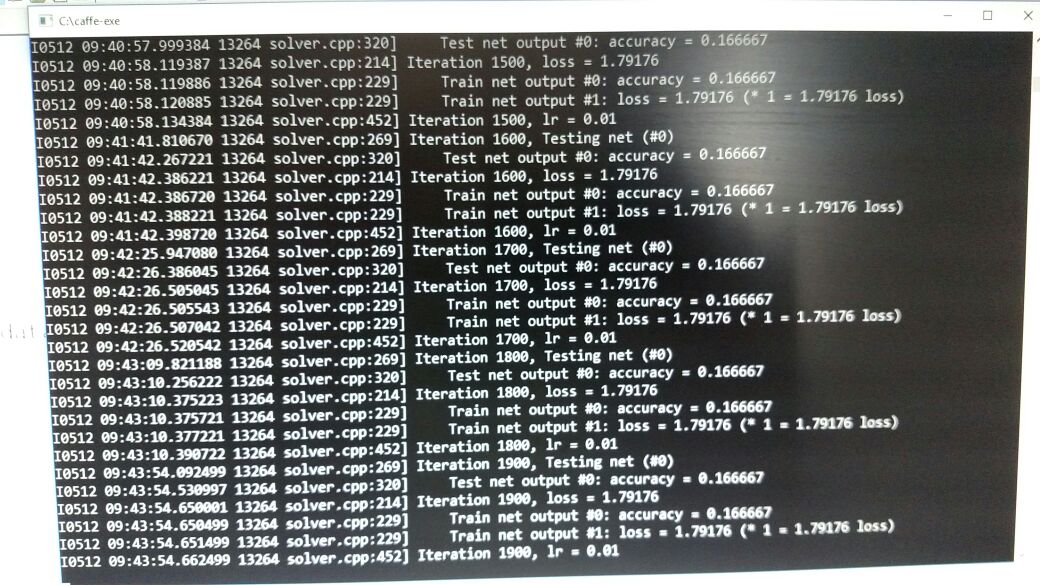


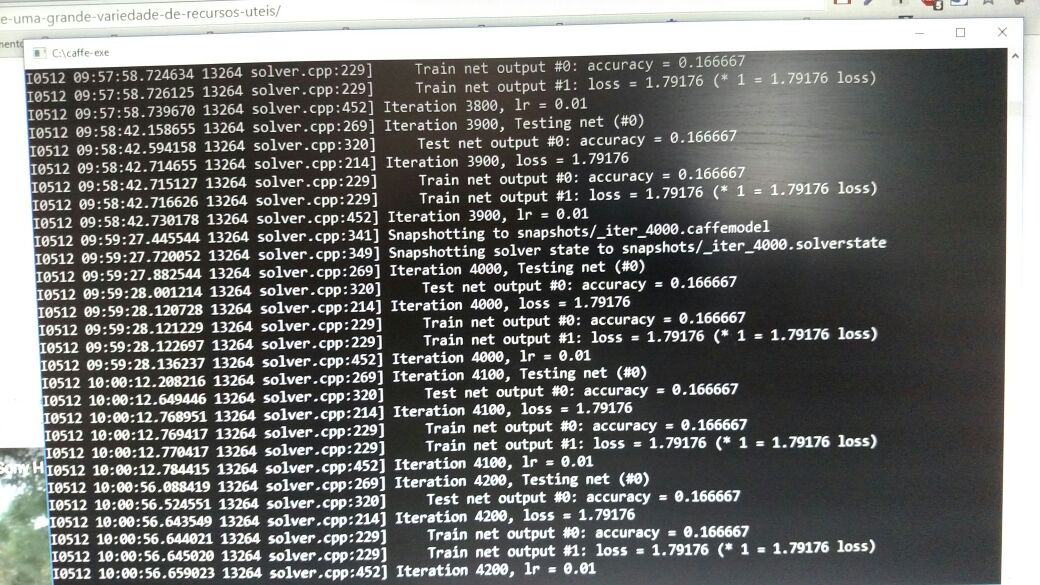


**Data :** 12/05/2016 **(Ambiente Windows) – Vários Processamentos durante o dia**

**Resultado :** Entendeu-se que a política para a taxa de aprendizagem do tipo INV poderia estar causando o rápido crescimento da acurácia para 20% mas depois decaindo até uma baixíssima a taxa de aprendizagem o que impediria o crescimento da acurácia a partir dos 20% (ou seja mantendo-a em um platô). Mudou-se a política para a taxa de aprendizagem para tipo fixa “FIXED”, conforme figura a seguir, porém, após 9.400 iterações da rede a acurácia que havia se iniciado em 16,66% se manteve neste valor até o fim (ver também copia da tela a seguir).

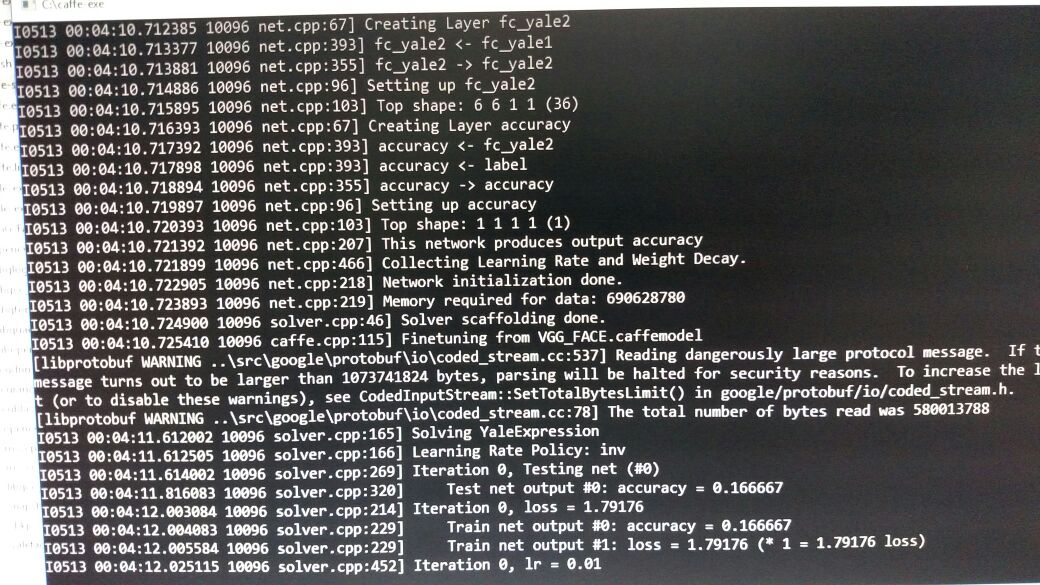


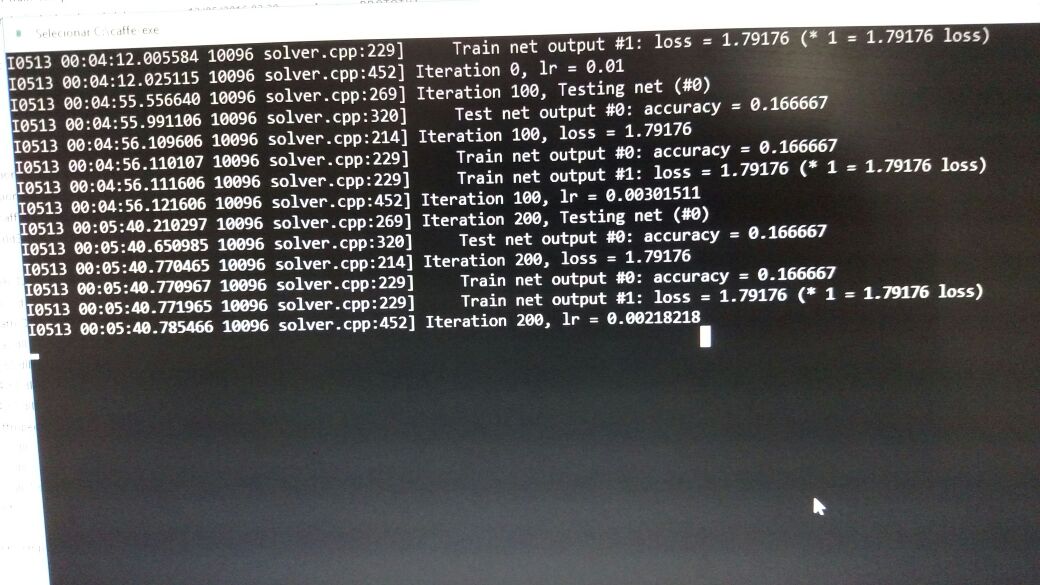




**Data :** 13/05/2016 **(Ambiente Windows) – Vários Processamentos durante o dia**

**Resultado :** Imaginou-se que o que poderia estar impactando o crescimento da acurácia pudesse ser a composição dos arquivos de teste e treino (que estavam respectivamente com 10% e 90% do conjunto de imagens do database de imagens que se estava utilizando ) e, por isto, novas composições de arquivos de teste e treino foram processadas (30% de teste e 70% de treino e 50% de teste e 50% de treino) sem que se obtivesse melhoria da acurácia. Como exemplo, mostra-se a seguir cópias de telas de processamentos feitos sem aumento da acurácia)



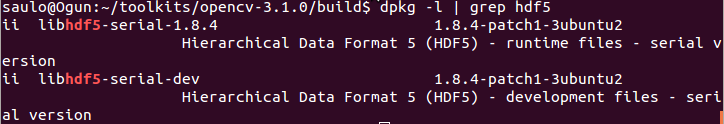


**Data :** 14/05/2016 : Decisão por instalar a Versão da Caffe para o Ubuntu 12.04 no computador que se estava usando para recomeçar a processar a rede nesse sistema operacional e, como plano B, tentar processar a rede em um dos computadores mais potentes do LCAD.

**Período** : 16 a 19/05 – Instalação da Versão da Caffe para o Ubuntu 12.04 no computador que se estava usando para recomeçar a processar a rede nesse sistema operacional e testes.

Durante a instalação do novo ambiente no sistema Operacional Ubuntu 12.04, uma série de problemas foram aparecendo. Para ilustração dos problemas ocorridos segue-se a descrição de alguns deles.

* Não se conseguia baixar a biblioteca ippicv\_linux20151201 diretamente pelo procedimento de instalação e a solução foi de baixá-la diretamente do site;
* Foi encontrado erro no arquivo hdf5.cpp.o e após a verificação de existência através do comando dpkg -l | grep hdf5, foi identificada a sua versão (1.8.4) e a solução foi a de retirar o parâmetro de compilação do HDF (figura a seguir).



* Na instalação do OpenCV-master, obtido a partir do branch master do github, ao se executar o Cmake, foi apresentado o mesmo erro da falta da biblioteca ippicv\_linux20151201 e a solução adotada foi a mesma;
* Ao se executar o “make” não foi encontrada a classe reflection no protocol buffer do google. Verificou-se que ela existia no protocol buffer 3.0.0 betaV3, a solução foi então atualizar a versão do protocol buffer;
* Na compilação do protobuffer apareceu a necessidade do gmock-1.7.0 e a solução foi baixa-lo;
* Após a instalação do libtool para a build correta, foi necessário o autoconf para gerar o ./configure;
* A execução do protobuffer 3.0.0 falhou e, procurando nos fóruns, foi encontrada uma versão 2.5.0 recomendada como solução para o problema.

Após a instalação do ambiente Ubuntu no computador que estava sendo usado anteriormente, aconteceram vários problemas de “out of memory” e várias tentaivas de reconfiguração da rede (arquivo Net.Prototxt) foram feitas em relação à diminuição dos parâmetros batch-size (para diminuir o número de imagens carregadas na memória ao mesmo tempo, modificação do tamanho do kernel das camadas de convolução, etc.), até que o processamento ficou normalizado.

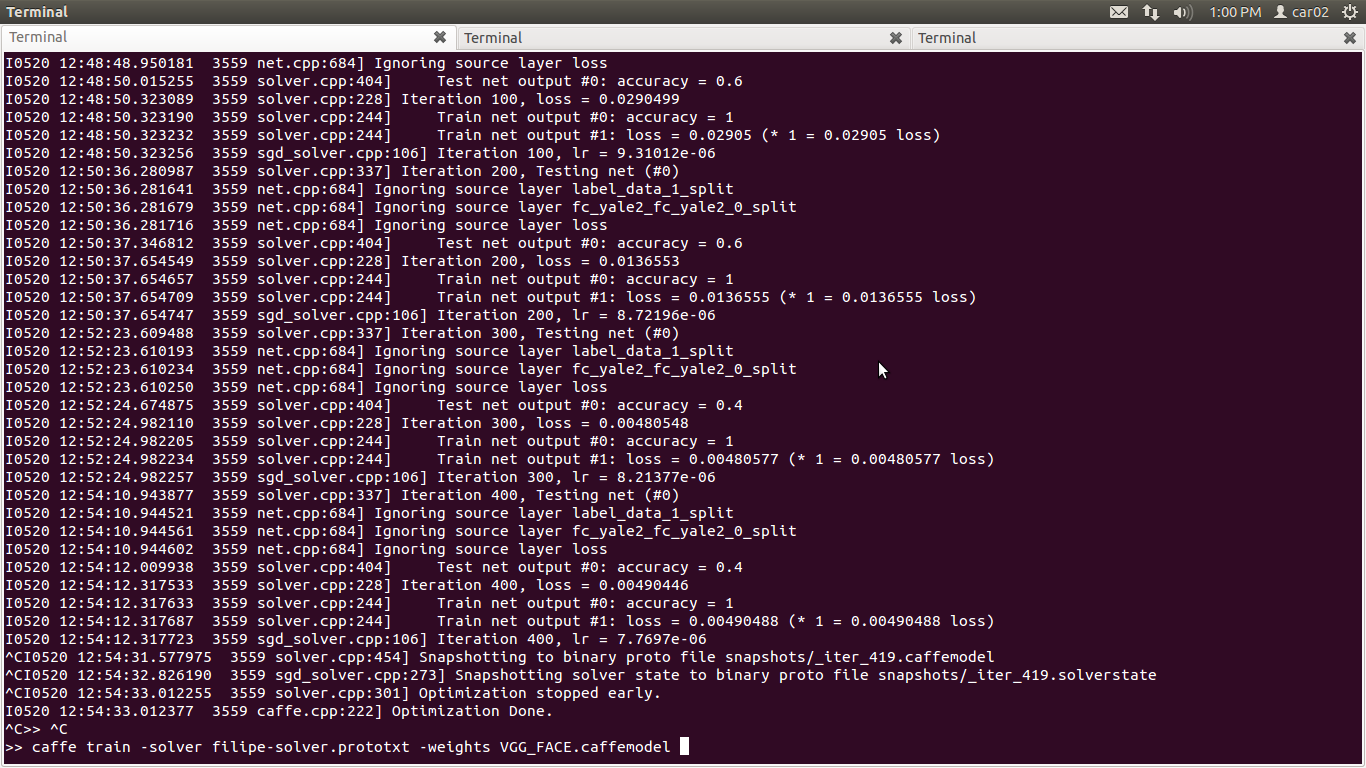
Após a normalização alguns processamentos foram feitos entretanto o tempo de processamento ficou muito alto inviabilizando o uso daquela instalação.

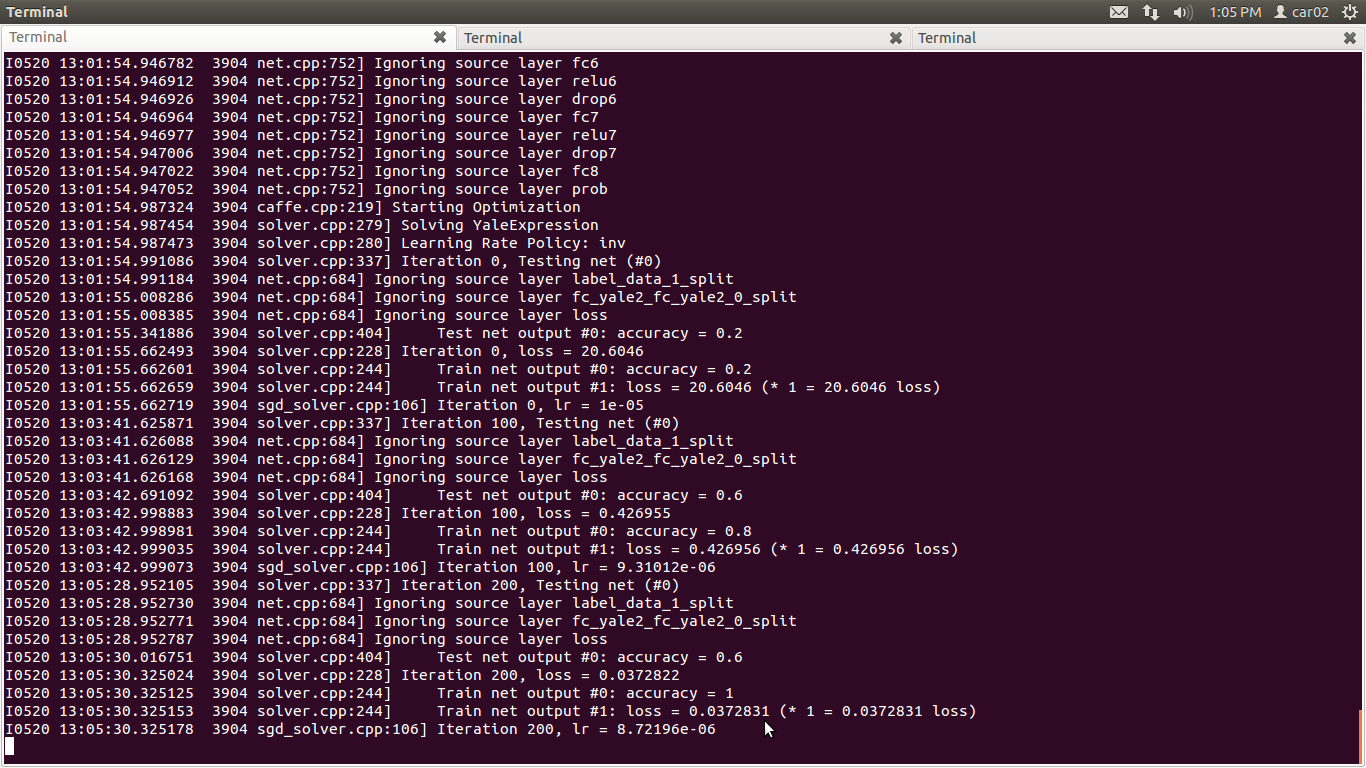
**Data : 20**/05/2016 : Processamento da Rede Cafee que já estava instalada em um computador do laboratório LCAD, mais potente em termos de recursos de hardware, e com sistema operacional Ubuntu 12.04 LTS, Open CV 2.4.9 e Cuda 7.

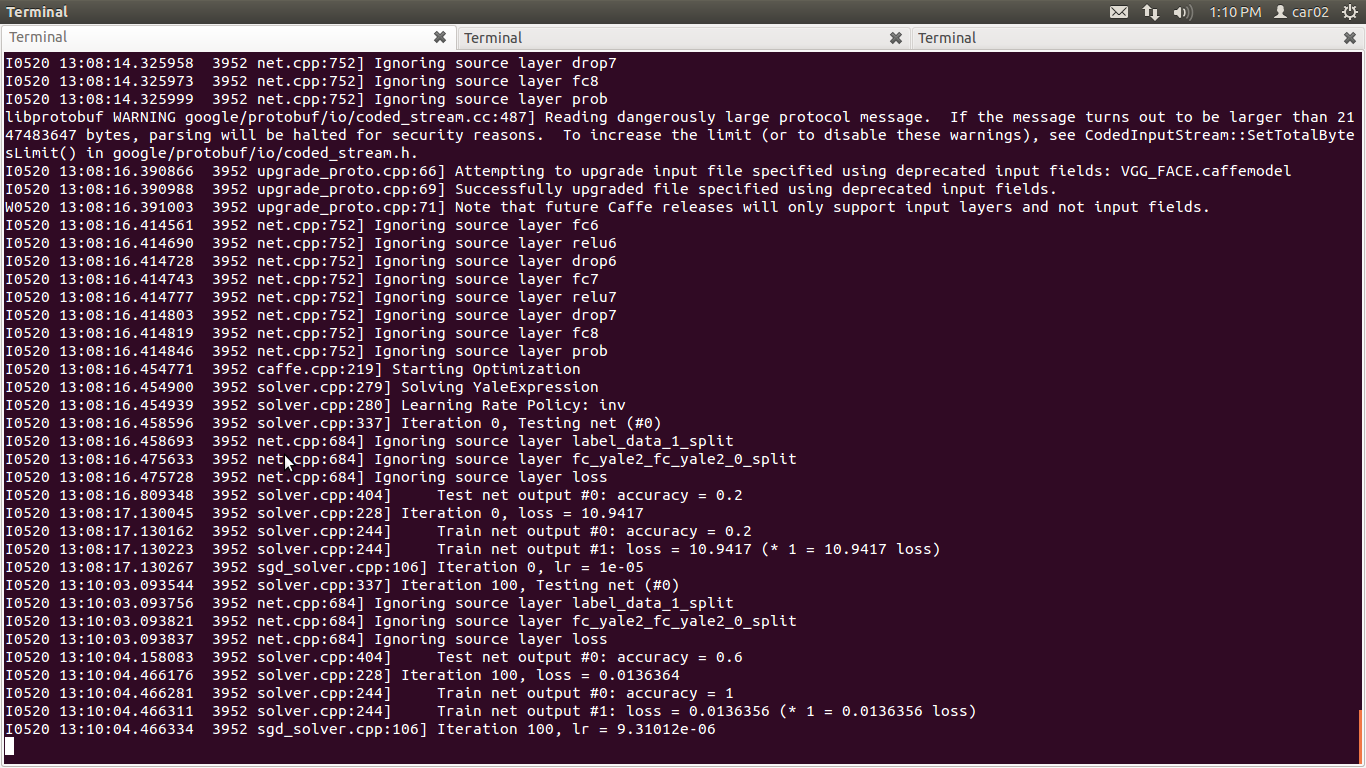
Tal computador tem a seguinte configuração: Processador Intel Xeon, Memória RAM de 12 GB e duas placas gráficas de 2 GB cada uma.

Neste processamento utilizou-se ainda a mesma configuração arquivos Solver.Prototxt e Net.Prototxt, que já estava sendo usada no computador anterior e com instalação Windows e, também , com o database de testes contem os 15 conjuntos de 6 imagens de uma mesma pessoa, cada uma imagem com uma expressão facial diferente.

**Resultado:** Utilizando-se esta nova instalação obteve-se um acurácia de 60% para testes e, várias vezes acurácia de 100% para treino.(Ver copias das telas com tais resultados a seguir).







**Data : 25**/05/2016

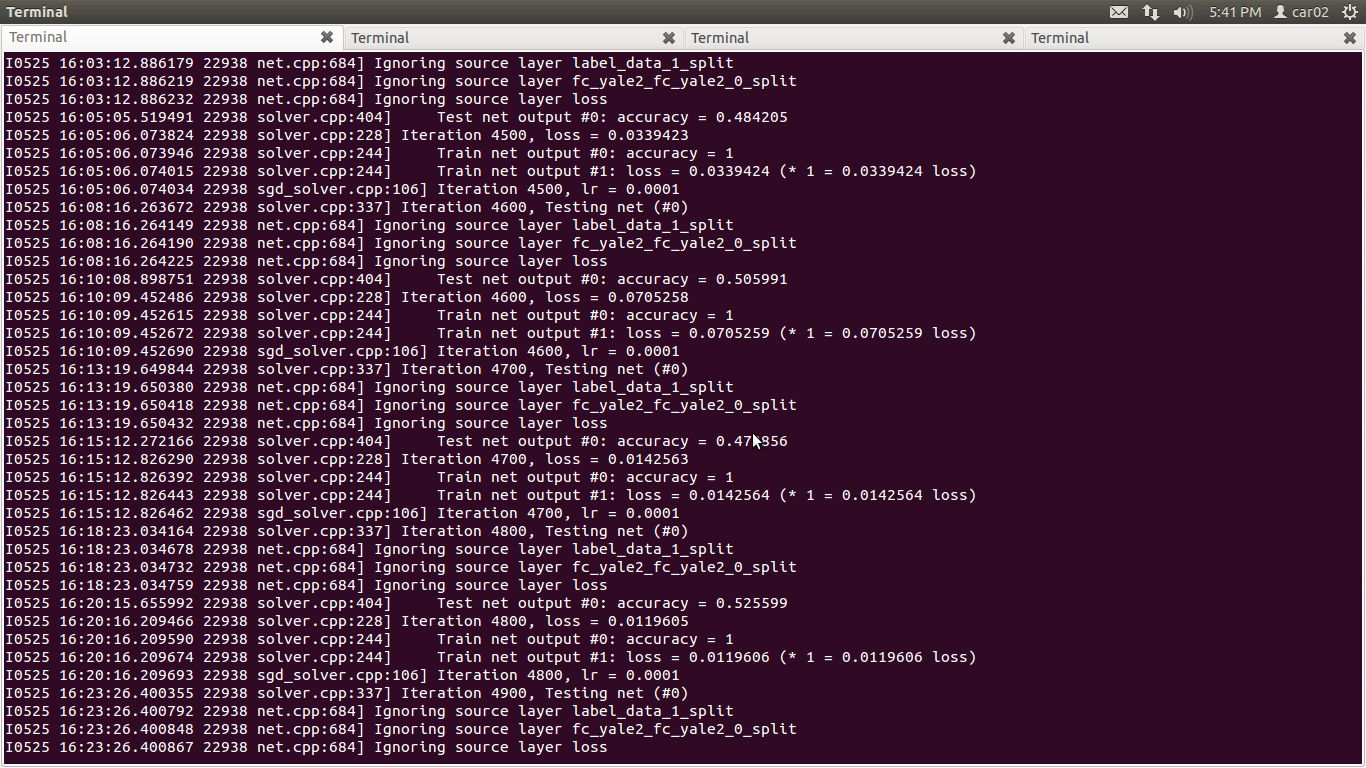
Inicialmente como se tratava de um processamento para obtenção da identificação das expressões faciais, fez-se uma depuração no database de imagens AR\_Face, que era composto por 76 conjuntos de imagens de homens e 60 conjuntos de imagens de mulheres, cada conjunto com 26 fotos da mesma pessoa em situações diferentes, eliminando aquelas fotos que não permitiam a correta identificação das faces (fotos com óculos, fotos com echarpes, fotos com brilho, etc.). Após essa depuração, os conjuntos ficaram compostos por 9 fotos por pessoa. Foram então criados os arquivos de teste com 10% das imagens disponíveis e de treino com 85% das imagens disponíveis.

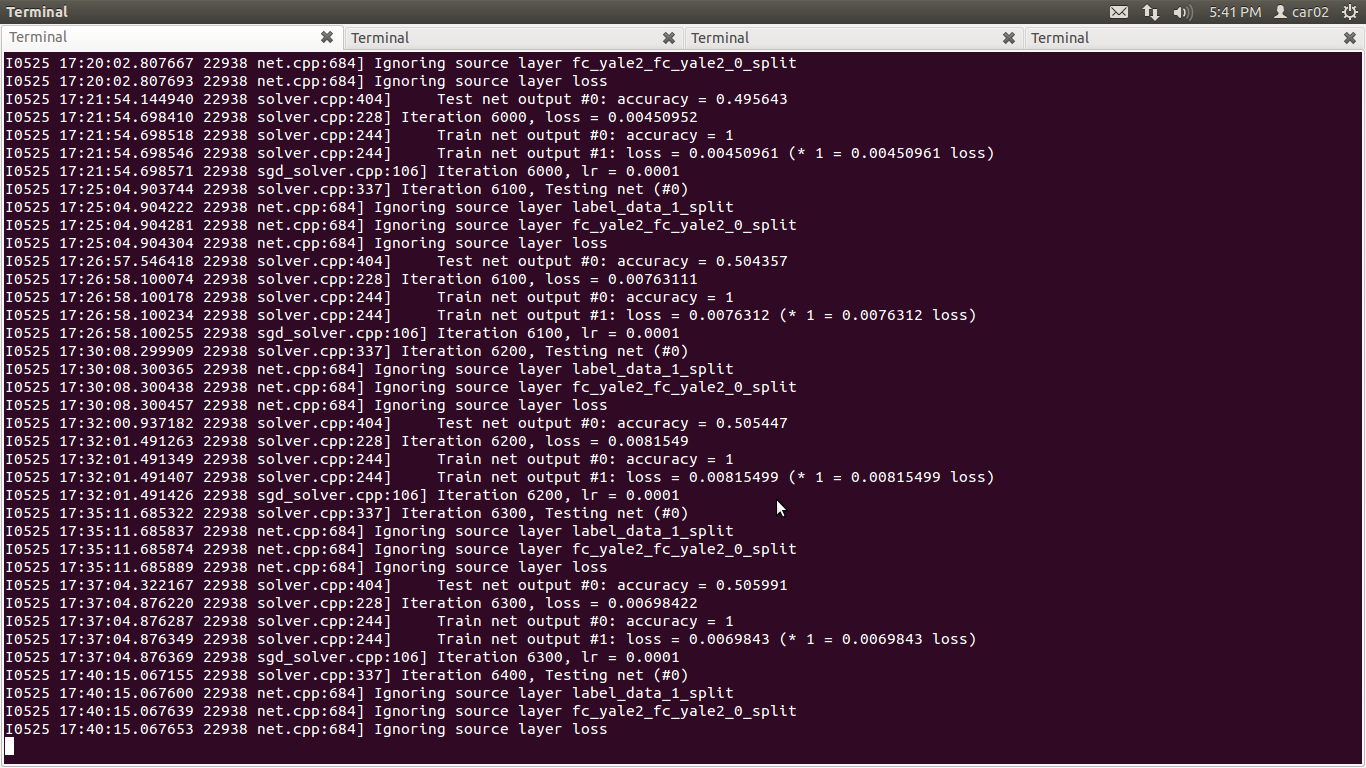
Utilizou-se a mesma configuração arquivos Solver.Prototxt e no arquivo da configuração da rede (Net.Prototxt) alterou-se:

* nas camadas de entrada de dados, o batch-size para 9 (pretendia-se se carregar apenas nove imagem de cada vez na memória para não se ter problema de out of memory e ter um processamento mais rápido), os parâmetros de altura e largura das imagens e os nomes dos arquivos de treino e teste;
* nas camadas de convolução foram alterados o no parâmetros blobs\_lr do que estavam (0.1 e 0.2 respectivamente ) para que essas camadas passassem a contribuir com os ajustes na qualidades das imagens.
* nas camadas de classificação foi alterado o output para 9 (nove tipos de expressões possíveis);

Foi feito então o Processamento da Rede Cafee que já estava instalada em um computador do laboratório LCAD.

**Resultado:** Uma acurácia de 52,55% para testes e, várias vezes acurácia de 100% para treino.(Ver copias das telas com tais resultados a seguir).





**Data : 01**/06/2016

Verificou-se que na primeira depuração do database AR\_Face a escolha de 9(nove) imagens como representativas de nove expressões foi equivocada pois deveriam ser apenas 4, uma se diferenciava apenas por ter brilho e as outras oito, duas a duas eram muito parecidas. Então optou-se para eliminar uma imagem de cada conjunto e classificar duas as duas como representativas da mesma expressão. O database de imagens foi então acertado.

Optou-se também por crescer o número de imagens dos arquivos de teste para 25% do database e o arquivo de treino passou para 75% do database.

Uma série de ajustes forma feitos nos arquivos de configuração do processamento (Solver. Prototxt) e no de configuração da rede (Net.Prototxt), a saber:

Solver. Prototxt:

* **test\_iter = passou-se de 1 para 55** (pois com 4 expressões o batch size deveria passar de 9 para 4 e, segundo as orientações dos pesquisadores da Caffe a multiplicação do test\_iter pelo batch\_size deveria ser igual ao número de imagens testadas)
* **lr\_policy = passou de "inv" para “fixed” para que não houvesso o rápido decaimento da taxa de aprendizagem**

no arquivo da configuração da rede (Net.Prototxt) alterou-se:

* nas camadas de entrada de dados, o batch-size para 4 ;
* nas camadas de classificação foi alterado o output para 4 (nove tipos de expressões possíveis);

Foi feito então o Processamento da Rede Cafee que já estava instalada em um computador do laboratório LCAD.

**Resultado: Obteve-se então uma acurácia de** urácia de 83,66% para testes e, várias vezes acurácia de 100% para treino.(Ver copias das telas com tais resultados a seguir).

